

פרק 5

שכבת הרשת

Network Layer

נערך ע"י ורדה גל.

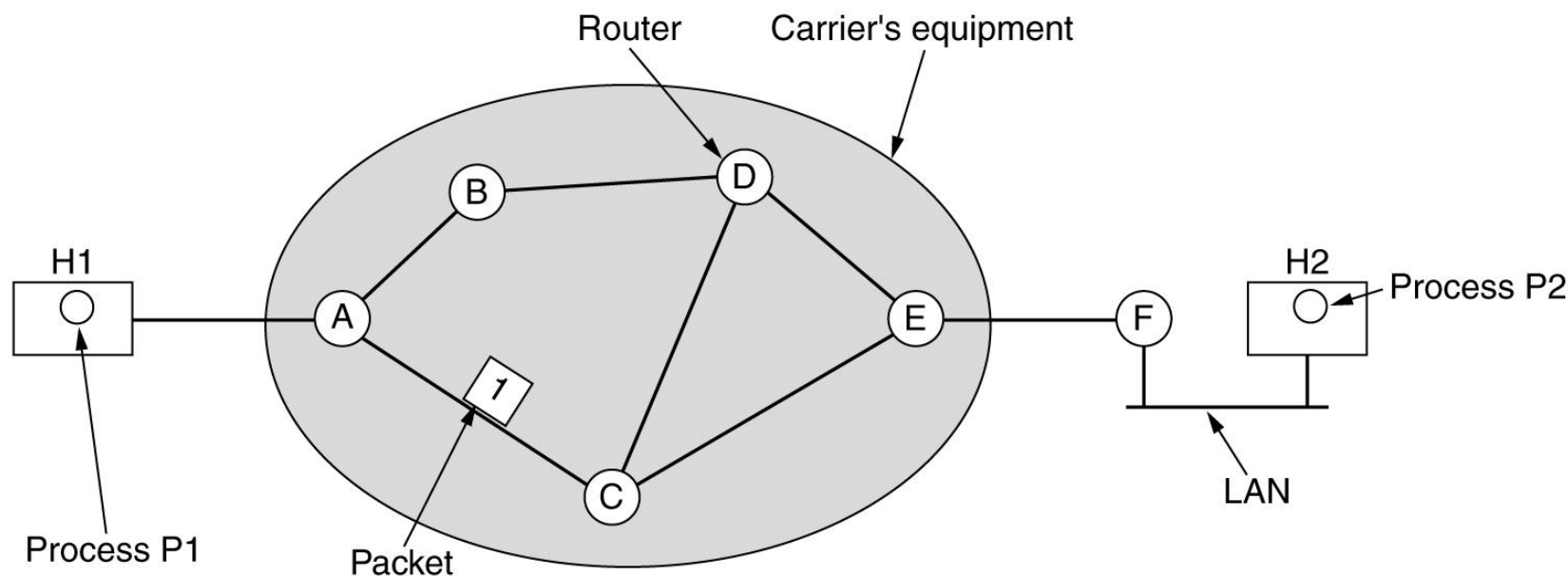
האמור במצגת זו אינו מהווה חלק רשמי מחומר הקורס.

רק האמור בספר הלימוד מחייב.

תפקידי שכבת הרשת:

- ניתוב.
- בקרת עומס.

- שכבת הרשת צריכה להעביר מנות מתחנת מקור לתחנת יעד – גם אם צריך לעבור תחנות ביניים.



Store-and-forward

שכבת הרשת יכולה לתת

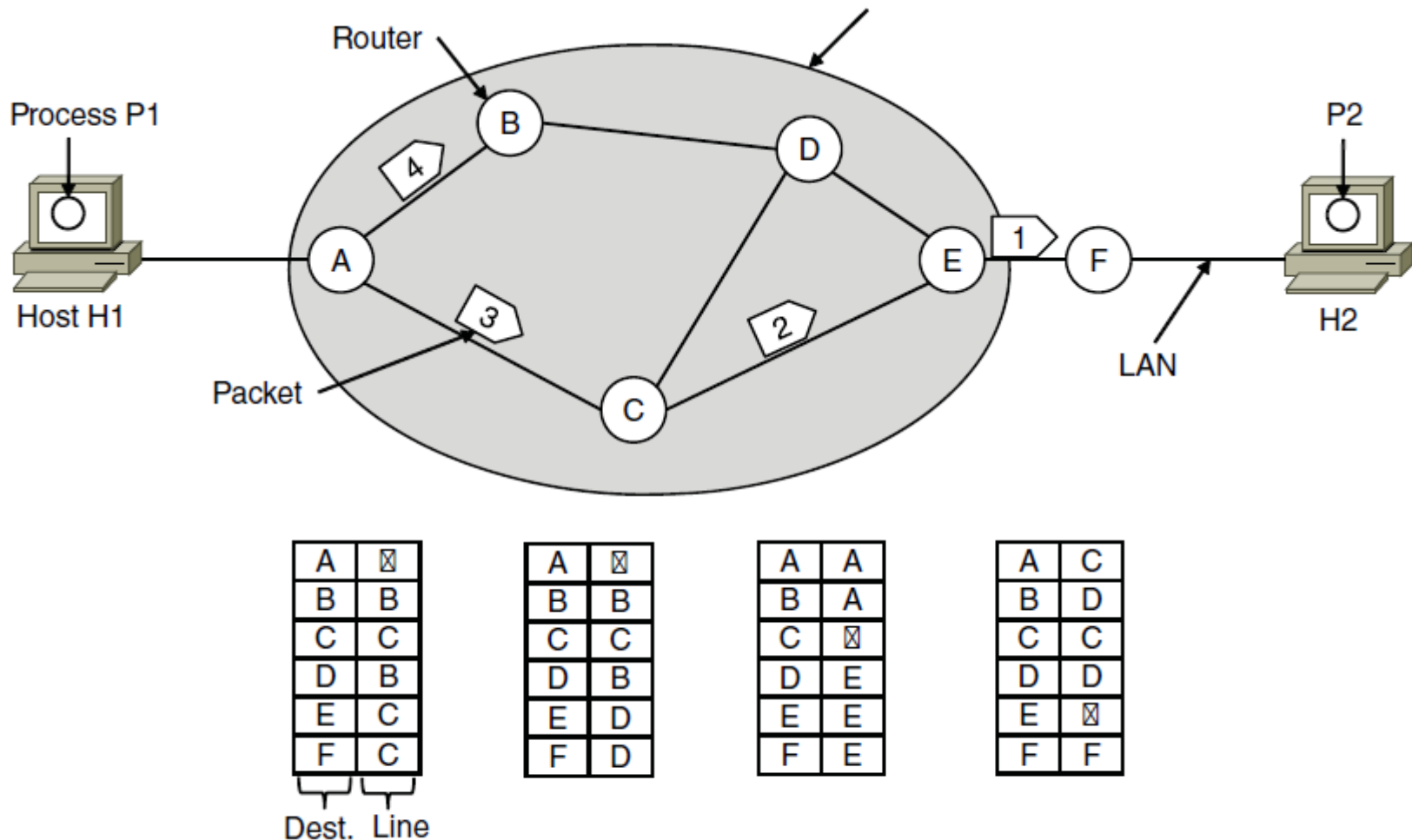
- שרותים מקושרים

או

- שרותים לא מקושרים.

מבוא לרשתות תקשורת מחשבים – פרק 5

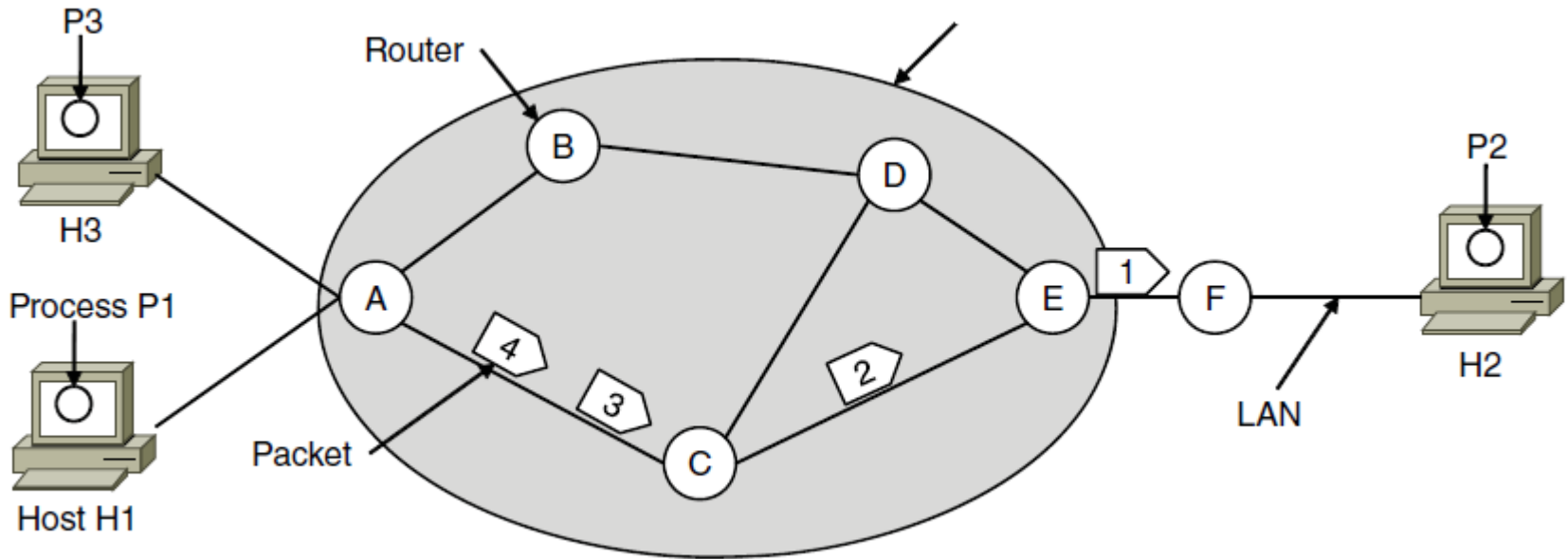
- שרות לא מקושר – תת רשת מברקים



Routing within a datagram subnet.

מבוא לרשתות תקשורת מחשבים – פרק 5

• שרות מקושר – תת-רשת מעגלים מדומים



| | | | | | | | | | | | |
|----|---|-----|---|---|---|---|---|---|---|---|---|
| H1 | 1 | C | 1 | A | 1 | E | 1 | C | 1 | F | 1 |
| H3 | 1 | C | 2 | A | 2 | E | 2 | C | 2 | F | 2 |
| In | | Out | | | | | | | | | |

Routing within a virtual-circuit subnet.

מבוא לרשתות תקשורת מחשבים – פרק 5

• מעגלים מדומים מול תת-רשת מברקים

| Issue | Datagram subnet | Virtual-circuit subnet |
|---------------------------|--|--|
| Circuit setup | Not needed | Required |
| Addressing | Each packet contains the full source and destination address | Each packet contains a short VC number |
| State information | Routers do not hold state information about connections | Each VC requires router table space per connection |
| Routing | Each packet is routed independently | Route chosen when VC is set up; all packets follow it |
| Effect of router failures | None, except for packets lost during the crash | All VCs that passed through the failed router are terminated |
| Quality of service | Difficult | Easy if enough resources can be allocated in advance for each VC |
| Congestion control | Difficult | Easy if enough resources can be allocated in advance for each VC |

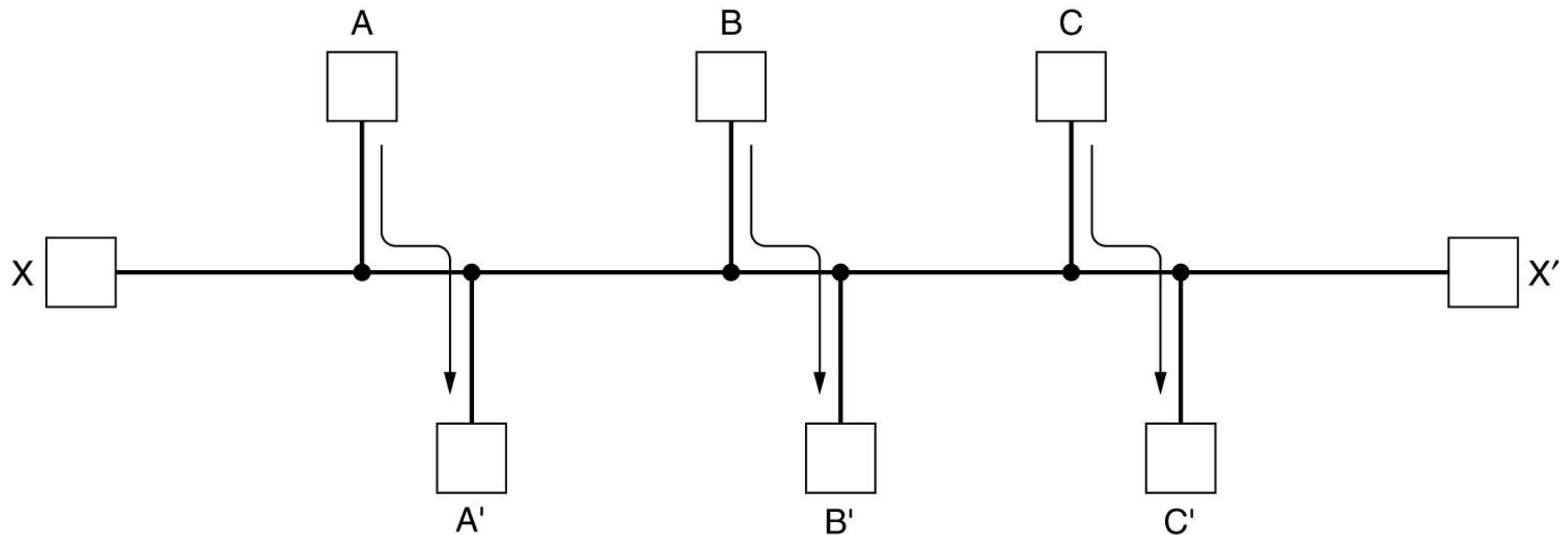
אלגוריתם הניתוב מחליט דרך איזה קו יציאה תשודר כל מנה. בתת-רשת מברקים, ההחלטה תעשה כל פעם מחדש. בתת-רשת מעגלים מדומים, ההחלטה תעשה פעם אחת בתחילת הקשר.

הדרישות מאלגוריתם ניתוב:

- נכונות
- יעילות
- עמידות בפני שינויים ברשת
- הוגנות
- אופטימליות

אלגוריתמים לניתוב

הוגנות מול אופטימליות:



אלגוריתמים לניתוב

אלגוריתמים סטטיים – ההחלטות נעשות מראש.

אלגוריתמים מסתגלים – משנים את ההחלטות
בהתאם לתעבורה ברשת ושינויים טופולוגיים.

אלגוריתמים לניתוב:

- עקרון האופטימליות
- המסלול הקצר ביותר (סטטי) - Shortest Path Routing
- הצפה (סטטי) – Flooding
- ניתוב וקטורי מרחק (דינמי) – Distance Vector Routing
- ניתוב מצב הקווים (דינמי) – Link State Routing
- ניתוב ברשתות ad hoc
- ניתוב מדרגי – Hierarchical Routing
- הפצה – Broadcast Routing
- ניתוב מרובה יעדים – Multicast Routing
- Anycast routing

עקרון האופטימליות

אם נתב K נמצא על המסלול האופטימלי לניתוב בין נתב I

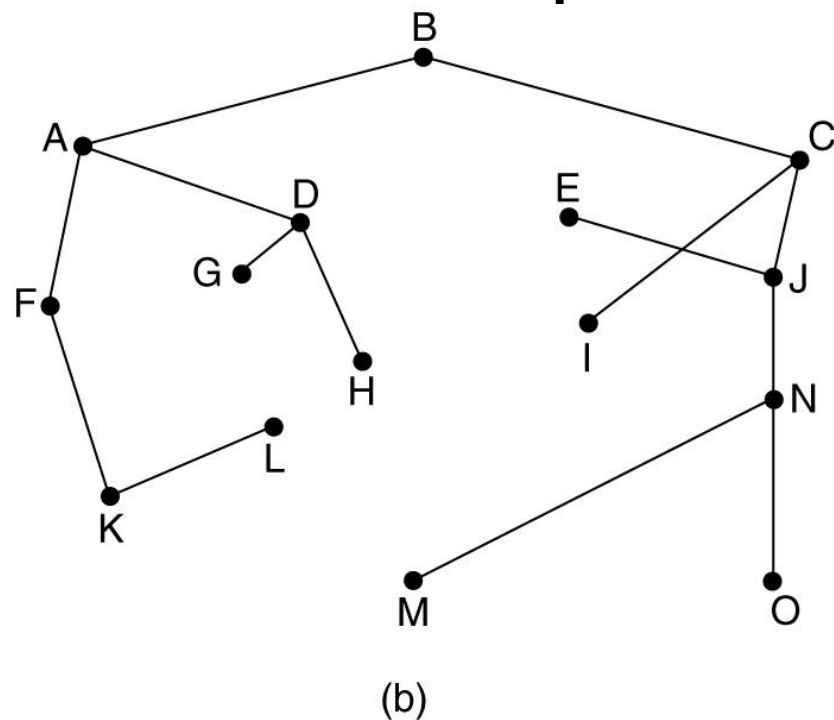
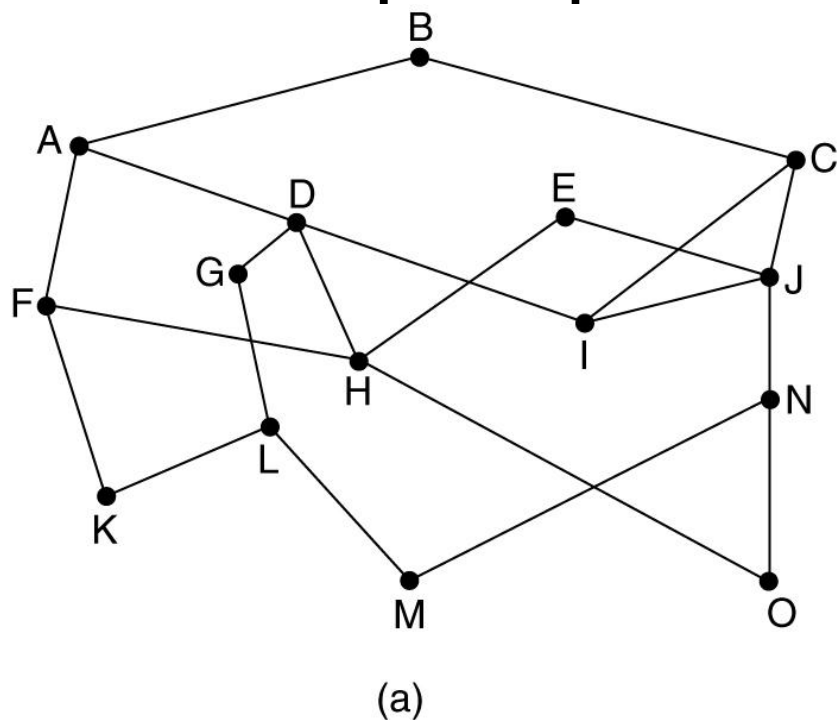
לנתב J , אזי הניתוב האופטימלי בין I ל- K ובין K ל- J

נמצא על אותו מסלול.

מבוא לרשתות תקשורת מחשבים – פרק 5

עץ יעד

אוסף המסלולים האופטימליים מכל המקורות ליעד מסוים, יוצר עץ ששורשו ביעד. העץ הזה נקרא עץ יעד.



(a) A subnet. (b) A sink tree for router B.

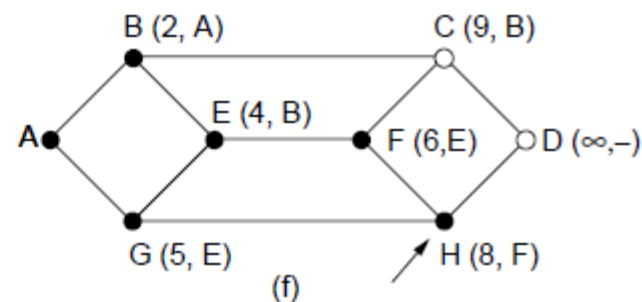
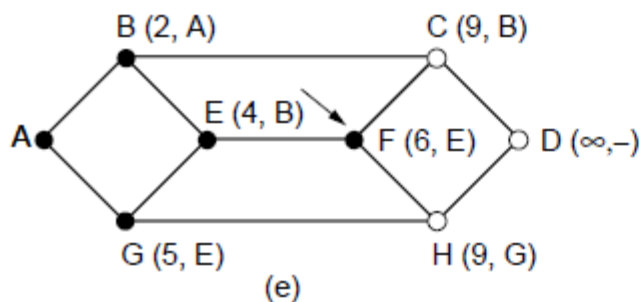
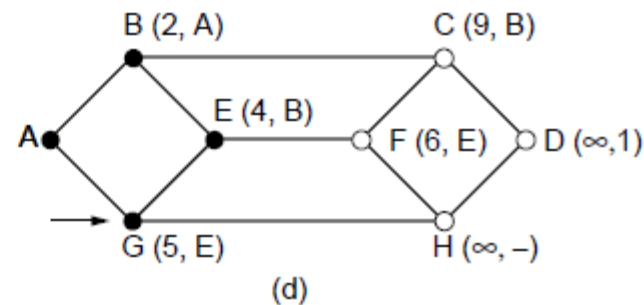
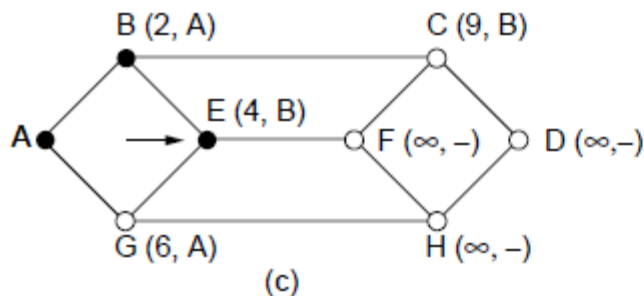
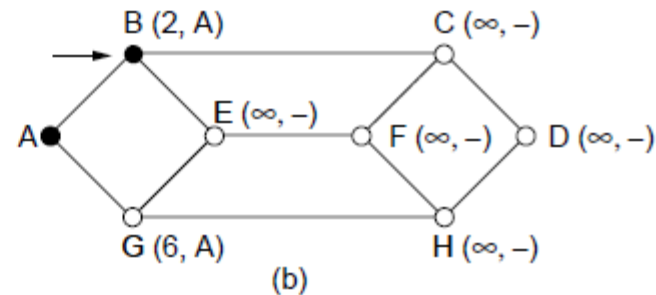
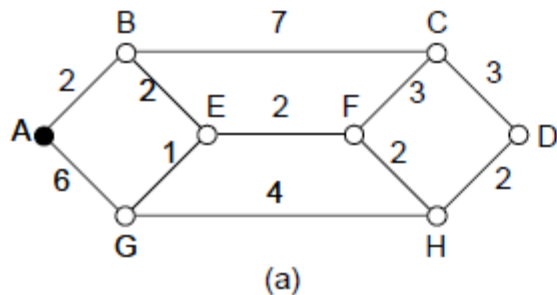
אורך מסלול יכול להיקבע ע"י פרמטרים שונים:

- מספר צמתי ביניים
- מרחק בקילומטרים
- השהיה ממוצעת על הקו כפי שנמדדה על מנות טסט
- רוחב הפס
- עומס תנועה
- עלות התקשורת
- אורך תור ממוצע

אפשר לקבוע משקל שהוא פונקציה של כל הפרמטרים האלה או חלקם, תוך מתן משקל מסוים לכל פרמטר.

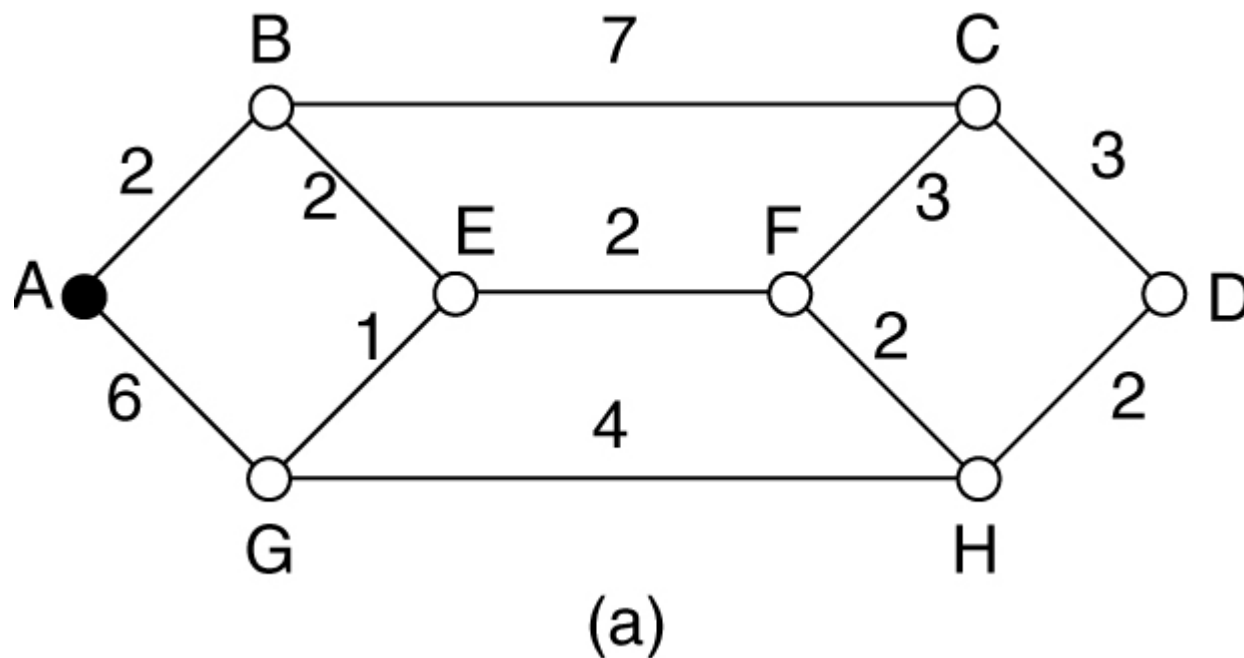
מבוא לרשתות תקשורת מחשבים – פרק 5

המסלול הקצר ביותר – האלגוריתם של דיקסטר (סטטי)
(מציאת המסלול הקצר ביותר מ-A ל-D)

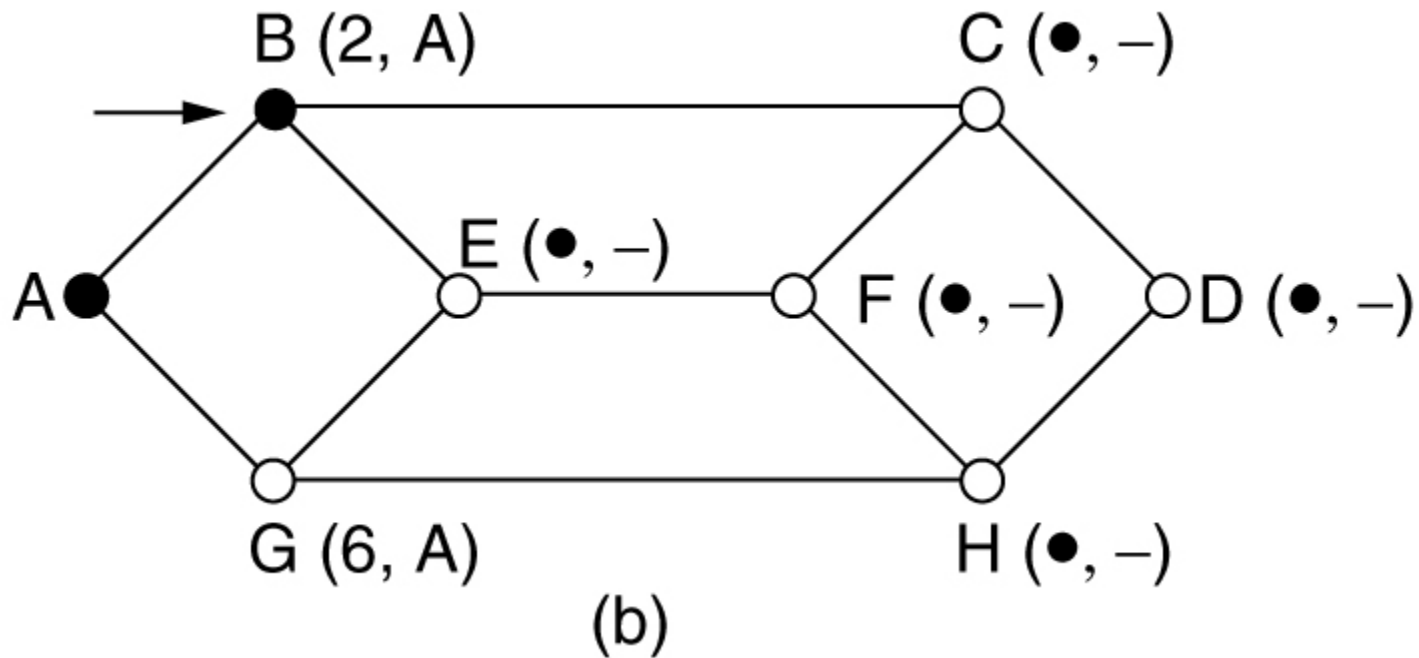


מבוא לרשתות תקשורת מחשבים – פרק 5

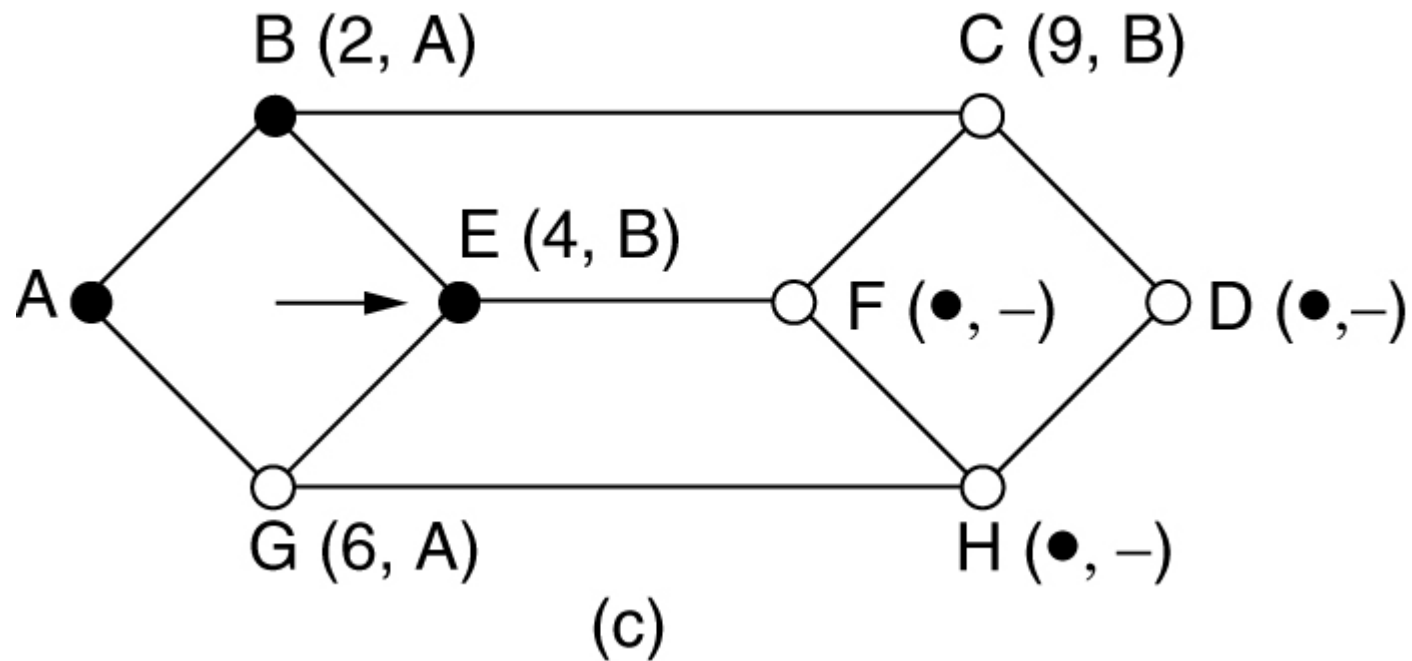
המסלול הקצר ביותר – האלגוריתם של דיקסטר
(מציאת המסלול הקצר ביותר מ-A ל-D)



המסלול הקצר ביותר – האלגוריתם של דיקסטר
(מציאת המסלול הקצר ביותר מ-A ל-D)

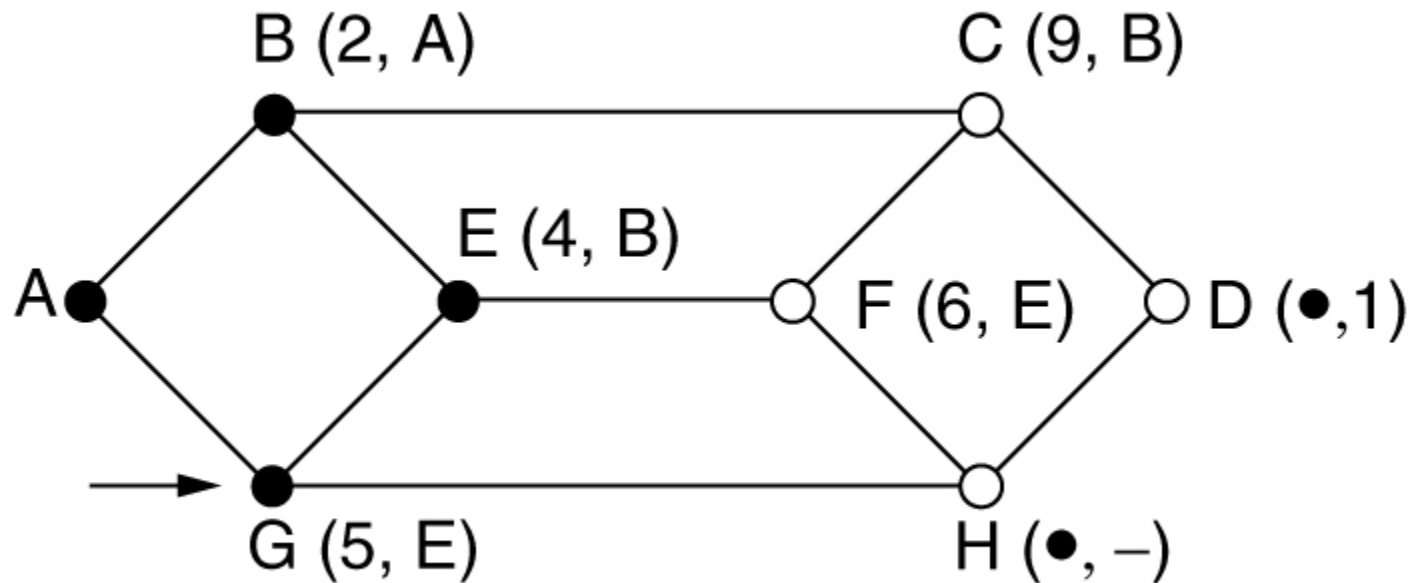


המסלול הקצר ביותר – האלגוריתם של דיקסטר
(מציאת המסלול הקצר ביותר מ-A ל-D)



מבוא לרשתות תקשורת מחשבים – פרק 5

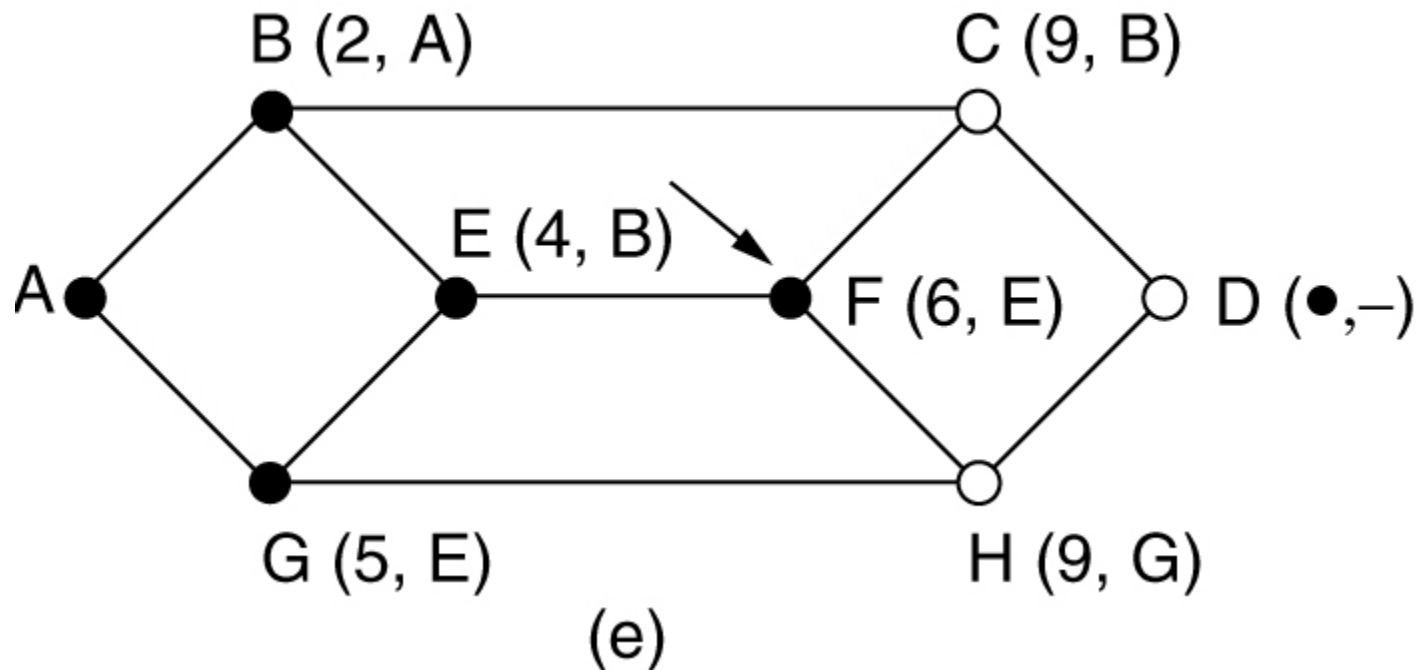
המסלול הקצר ביותר – האלגוריתם של דיקסטר
(מציאת המסלול הקצר ביותר מ-A ל-D)



(d)

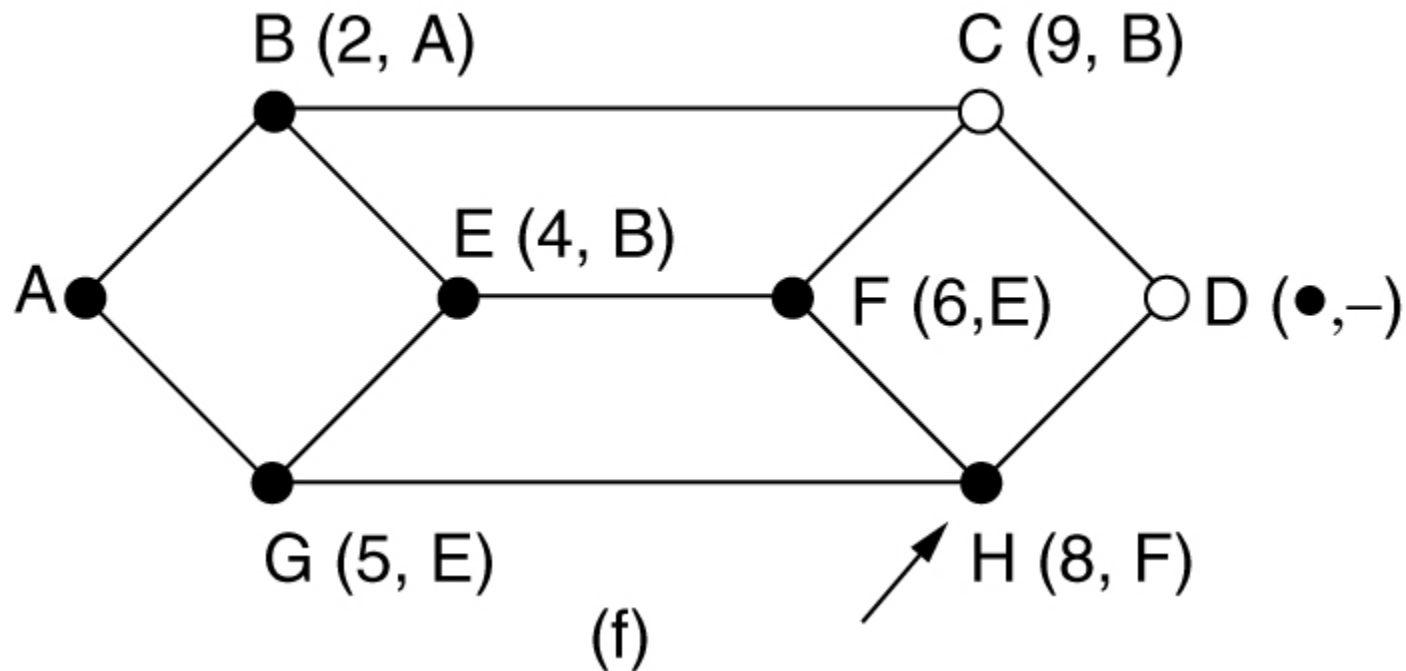
מבוא לרשתות תקשורת מחשבים – פרק 5

המסלול הקצר ביותר – האלגוריתם של דיקסטר
(מציאת המסלול הקצר ביותר מ-A ל-D)



מבוא לרשתות תקשורת מחשבים – פרק 5

המסלול הקצר ביותר – האלגוריתם של דיקסטר
(מציאת המסלול הקצר ביותר מ-A ל-D)



מבוא לרשתות תקשורת מחשבים – פרק 5

האלגוריתם של דיקסטר

```
#define MAX_NODES 1024          /* maximum number of nodes */
#define INFINITY 1000000000     /* a number larger than every maximum path */
int n, dist[MAX_NODES][MAX_NODES]; /* dist[i][j] is the distance from i to j */

void shortest_path(int s, int t, int path[])
{ struct state {                /* the path being worked on */
    int predecessor;            /* previous node */
    int length;                 /* length from source to this node */
    enum {permanent, tentative} label; /* label state */
} state[MAX_NODES];

int i, k, min;
struct state *p;

for (p = &state[0]; p < &state[n]; p++) { /* initialize state */
    p->predecessor = -1;
    p->length = INFINITY;
    p->label = tentative;
}
state[t].length = 0; state[t].label = permanent;
k = t;                /* k is the initial working node */
```

האלגוריתם של דיקסטר (2)

```
do {
    /* Is there a better path from k? */
    for (i = 0; i < n; i++)
        /* this graph has n nodes */
        if (dist[k][i] != 0 && state[i].label == tentative) {
            if (state[k].length + dist[k][i] < state[i].length) {
                state[i].predecessor = k;
                state[i].length = state[k].length + dist[k][i];
            }
        }

    /* Find the tentatively labeled node with the smallest label. */
    k = 0; min = INFINITY;
    for (i = 0; i < n; i++)
        if (state[i].label == tentative && state[i].length < min) {
            min = state[i].length;
            k = i;
        }
    state[k].label = permanent;
} while (k != s);

/* Copy the path into the output array. */
i = 0; k = s;
do {path[i++] = k; k = state[k].predecessor; } while (k >= 0);
}
```

הצפה (אלגוריתם סטטי)

שולחים את המנה בכל הקווים היוצאים, חוץ מהקו שדרכו היא התקבלה.

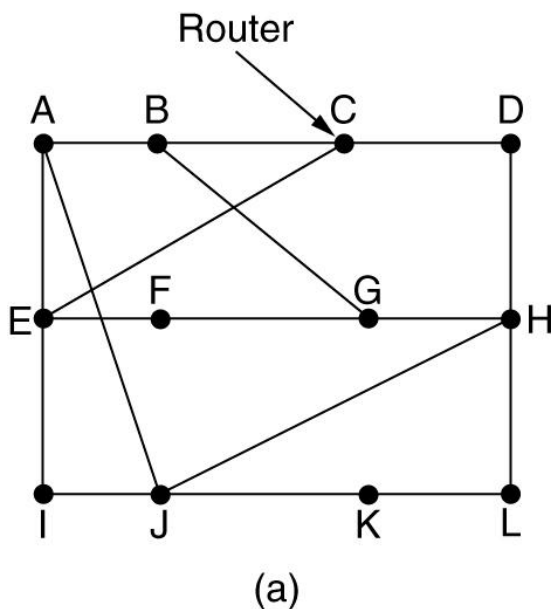
איך למנוע מצב שבו מנה תסתובב ברשת עד אין-סוף?

הצפה סלקטיבית

שולחים את המנה רק בקווים היוצאים שמקדמים אותה למטרה.

הצפה תכלול תמיד גם את המסלול הקצר ביותר.

ניתוב וקטורי מרחק – Distance Vector Routing



New estimated delay from J

| To | A | I | H | K | Line |
|----|----|----|----|----|------|
| A | 0 | 24 | 20 | 21 | 8 A |
| B | 12 | 36 | 31 | 28 | 20 A |
| C | 25 | 18 | 19 | 36 | 28 I |
| D | 40 | 27 | 8 | 24 | 20 H |
| E | 14 | 7 | 30 | 22 | 17 I |
| F | 23 | 20 | 19 | 40 | 30 I |
| G | 18 | 31 | 6 | 31 | 18 H |
| H | 17 | 20 | 0 | 19 | 12 H |
| I | 21 | 0 | 14 | 22 | 10 I |
| J | 9 | 11 | 7 | 10 | 0 – |
| K | 24 | 22 | 22 | 0 | 6 K |
| L | 29 | 33 | 9 | 9 | 15 K |

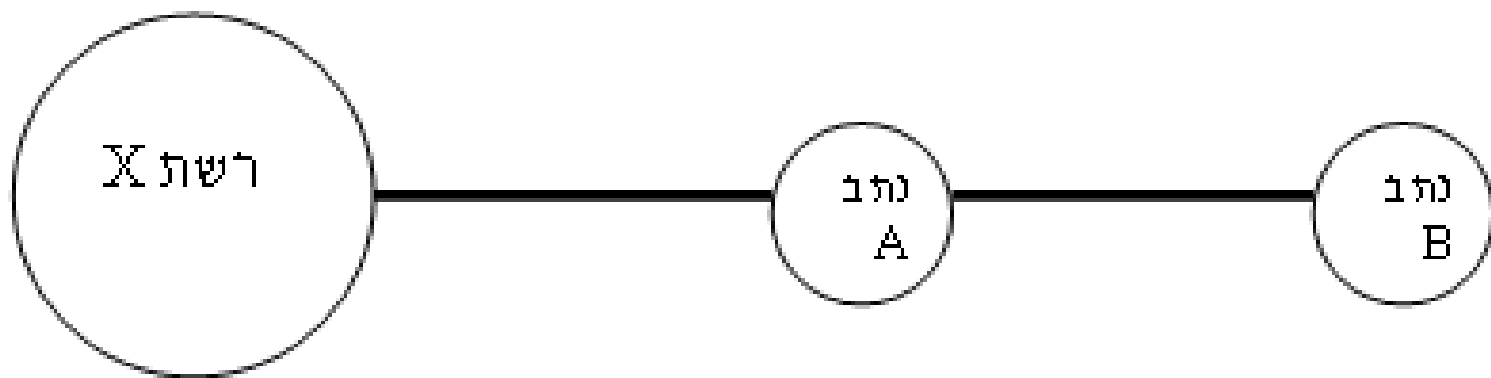
| | | | |
|---------------|----------------|----------------|---------------|
| JA delay is 8 | JI delay is 10 | JH delay is 12 | JK delay is 6 |
|---------------|----------------|----------------|---------------|

Vectors received from J's four neighbors

New routing table for J

הוקטורים שהגיעו מ-A, I, H, K
 וטבלת הניתוב המעודכנת של J.

בעית הספירה לאינסוף



ניתוב לפי מצב הקווים – Link State Routing

כל צומת אחראי לבדוק את מרחקו מכל אחד משכניו,
ולהפיץ מידע זה בלבד בכל הרשת.

כל צומת יכול אח"כ להפעיל את האלגוריתם של דייקסטרה
(או אחר) לחישוב המסלול הקצר ביותר לכל יעד ברשת.
כאן לא קיימת בעיית הספירה לאינסוף.

מבוא לרשתות תקשורת מחשבים – פרק 5

ניתוב לפי מצב הקווים – Link State Routing

השלבים:

1. לימוד אודות השכנים, ומה כתובותיהם ברשת.

2. מדידת מחיר הקו או ההשהיה לכל אחד מהשכנים.

3. בנית מנות מצב הקווים עם כל האינפורמציה שהוא למד –

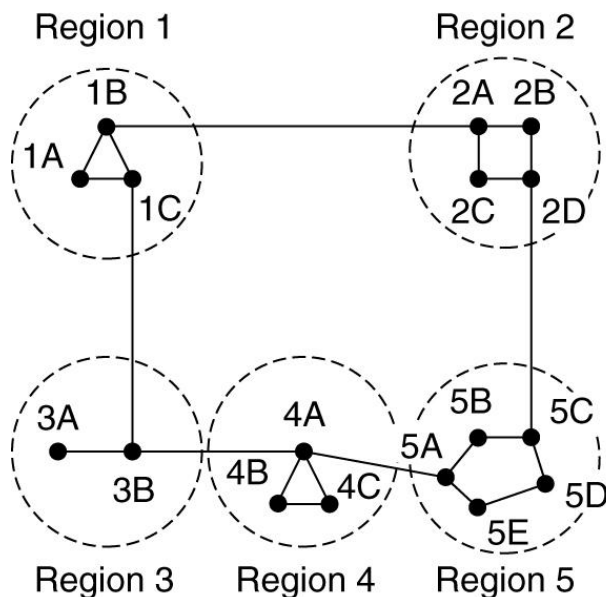
Link State Packets.

4. הפצת המנות/קבלת המנות של מצב הקווים לכל/מכל הצמתים

האחרים.

5. חישוב מסלולים חדשים קצרים ביותר לכל אחד מהצמתים האחרים.

ניתוב מדרגי – Hierarchical Routing



(a)

Full table for 1A

| Dest. | Line | Hops |
|-------|------|------|
| 1A | – | – |
| 1B | 1B | 1 |
| 1C | 1C | 1 |
| 2A | 1B | 2 |
| 2B | 1B | 3 |
| 2C | 1B | 3 |
| 2D | 1B | 4 |
| 3A | 1C | 3 |
| 3B | 1C | 2 |
| 4A | 1C | 3 |
| 4B | 1C | 4 |
| 4C | 1C | 4 |
| 5A | 1C | 4 |
| 5B | 1C | 5 |
| 5C | 1B | 5 |
| 5D | 1C | 6 |
| 5E | 1C | 5 |

(b)

Hierarchical table for 1A

| Dest. | Line | Hops |
|-------|------|------|
| 1A | – | – |
| 1B | 1B | 1 |
| 1C | 1C | 1 |
| 2 | 1B | 2 |
| 3 | 1C | 2 |
| 4 | 1C | 3 |
| 5 | 1C | 4 |

(c)

ניתוב באמצעות הפצה – Broadcast Routing

משתמשים כאשר רוצים לשדר מידע לכל צמתי הרשת.

האפשרויות:

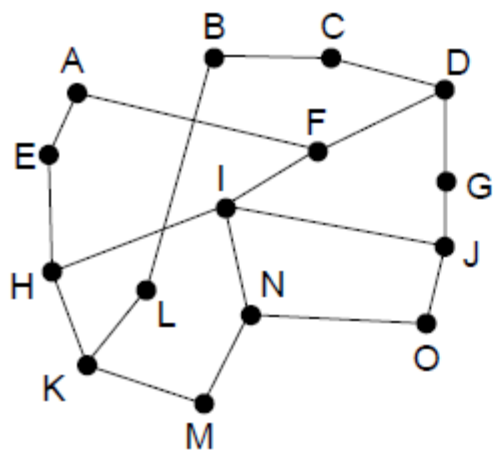
1. משלוח מנה לכל יעד – בזבזני ומחייב רשימה מלאה של כל היעדים.
2. הצפה – הרבה מנות, בזבוז של רוחב פס.
3. מנה עם רשימת יעדים, שתשוכפל בכל צומת – בזבוז.
4. לפי עץ פורש – כל הצמתים צריכים להסכים על אותו עץ.
5. הפצה על סמך המסלול ההפוך – Reverse Path Forwarding.

הפצה על סמך המסלול ההפוך –
Reverse Path Forwarding

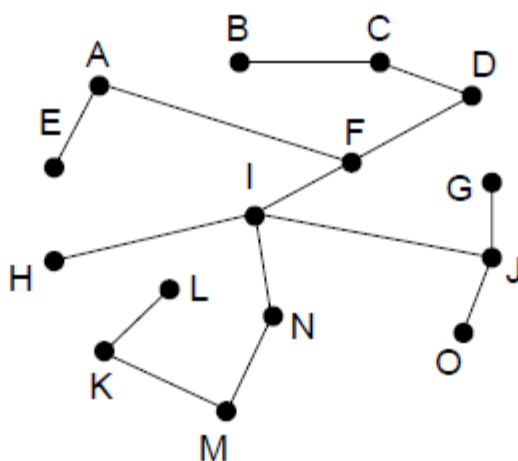
אם מנה מגיעה בקו המשמש לשליחת הודעות אל X ,

היא מועברת בכל שאר הקווים היוצאים.

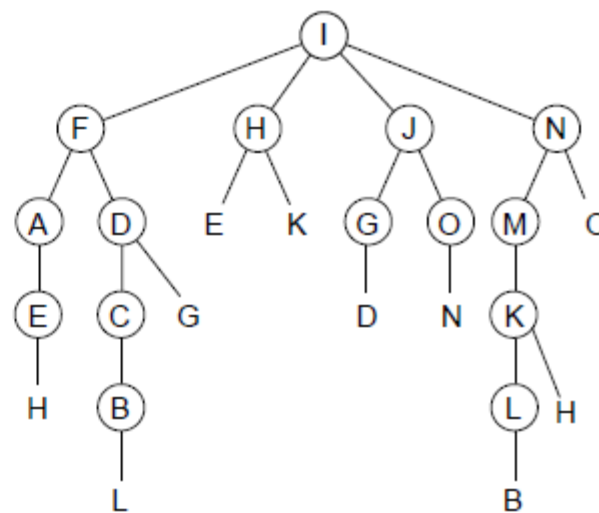
הפצה על סמך המסלול הפוך – Reverse Path Forwarding



(a)



(b)



(c)

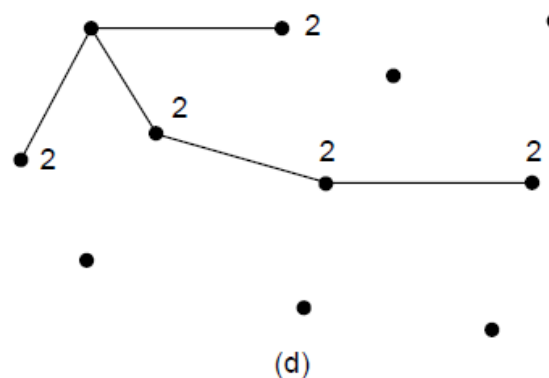
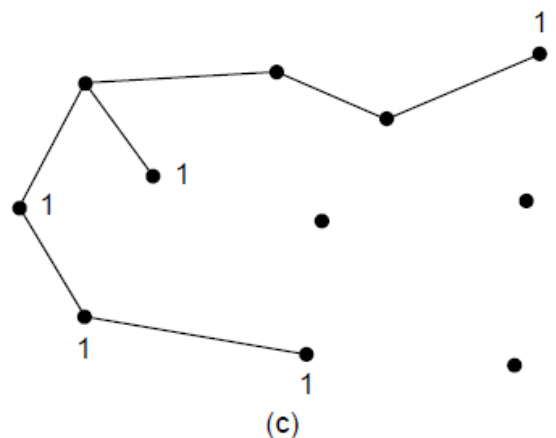
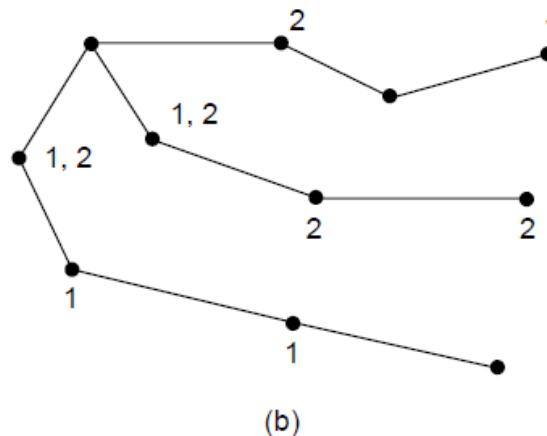
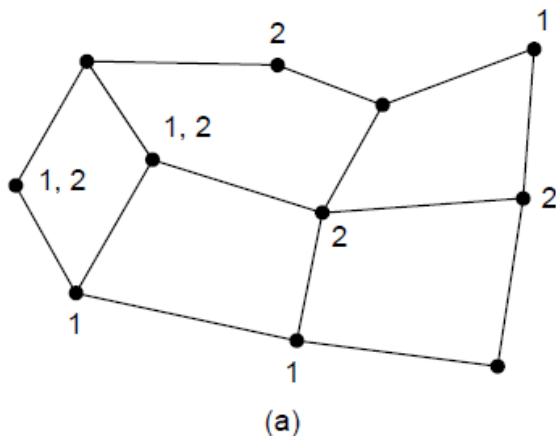
(a) תת-רשת.

(b) עץ פורש.

(c) העץ שנוצר ע"י הפצה ע"ס המסלול הפוך.

Multicast Routing – ניתוב מרובה יעדים

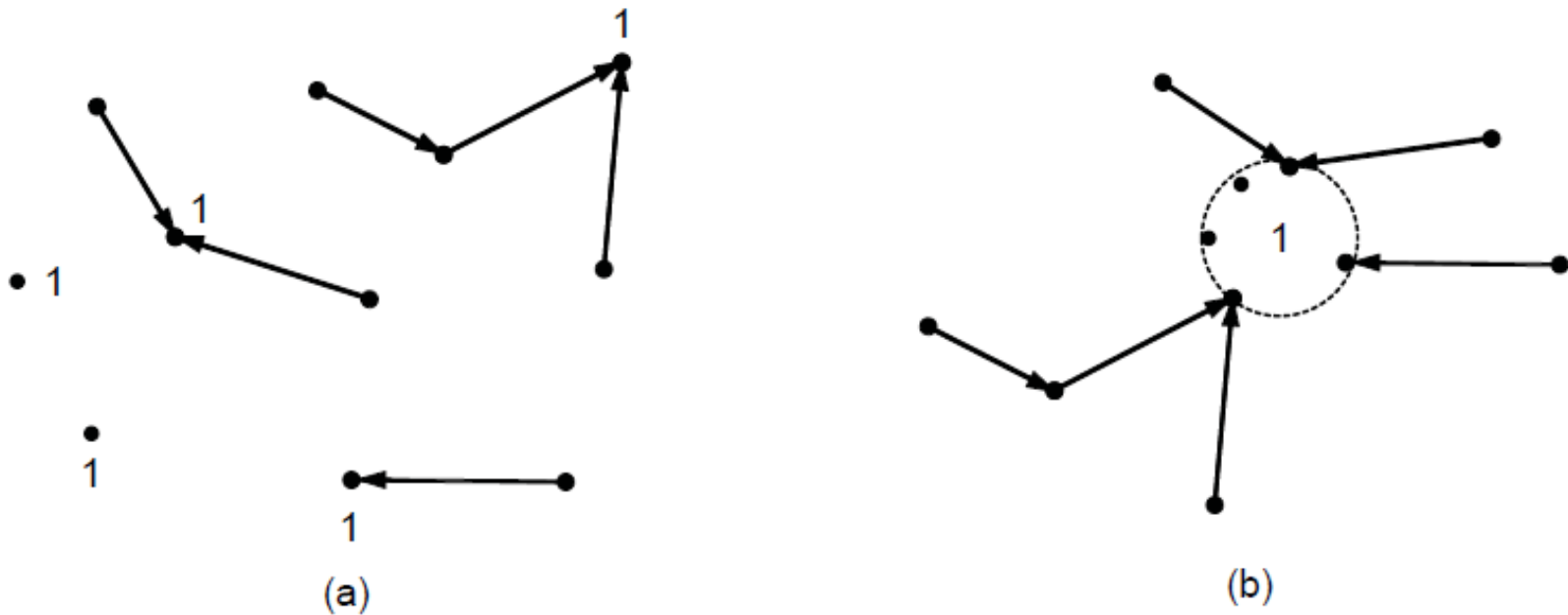
משלוח אותה הודעה לקבוצת יעדים, אך לא לכל הרשת.



(a) - רשת. (b) – עץ פורש עבור הנתב השמאלי. (c) – עץ מרובה יעדים עבור קבוצה 1. (d) – עץ מרובה יעדים עבור קבוצה 2.

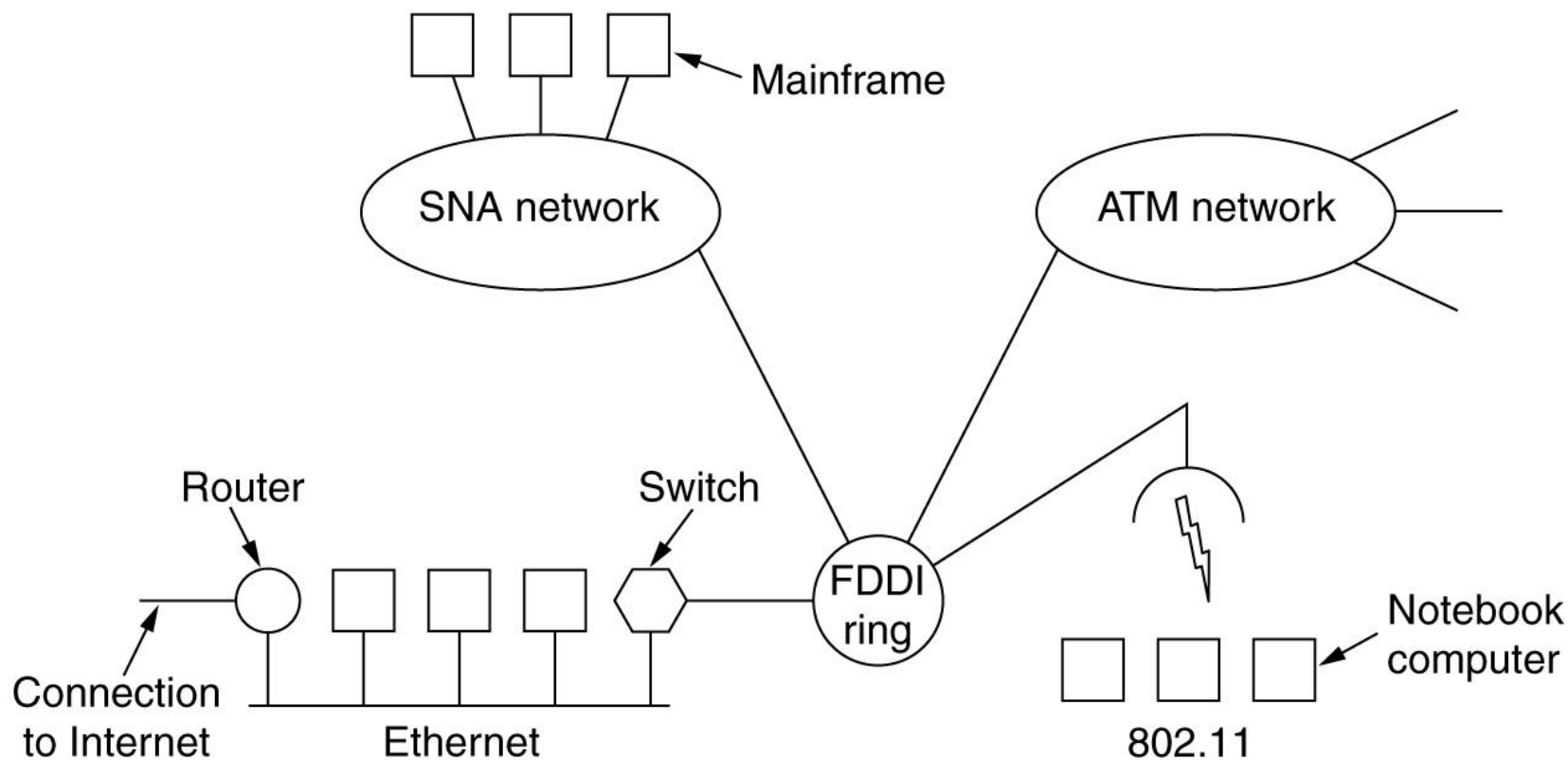
Anycast Routing

משלוח אותה הודעה לקבוצת יעדים, אך לא לכל הרשת.



- Anycast routes לקבוצה 1.
- טופולוגית הרשת כפי שהיא נראית לפרוטוקול הניתוב.

חיבור בין רשתות – Connecting Networks

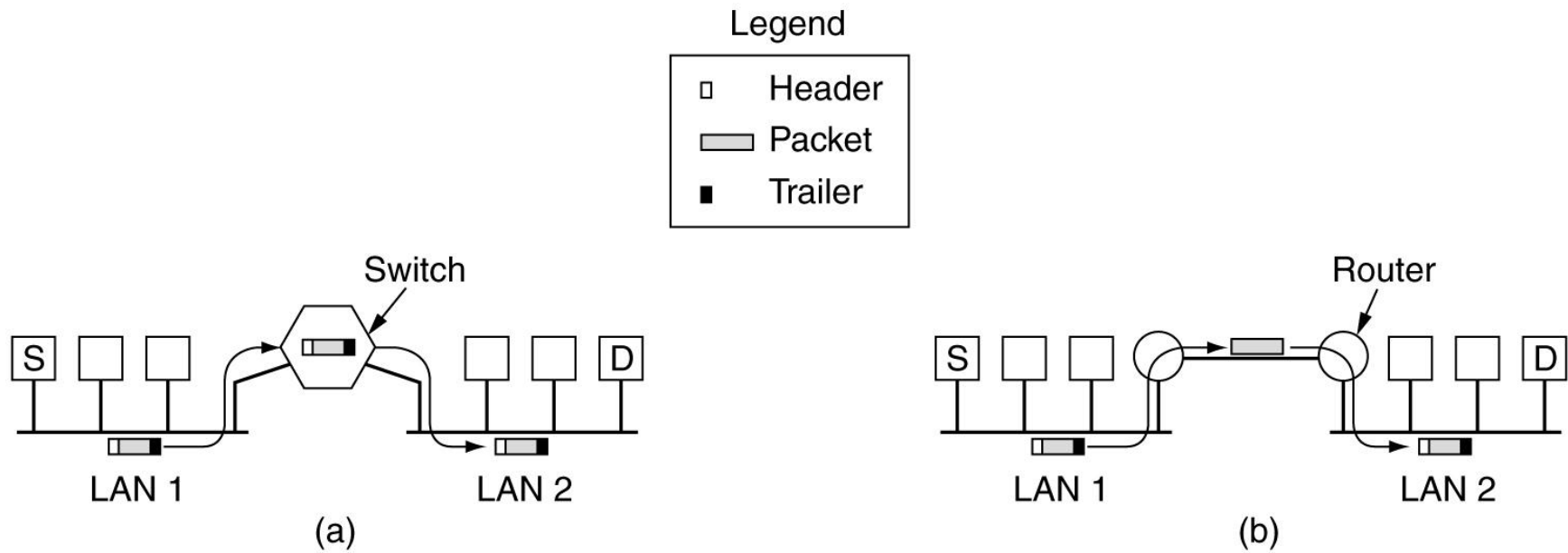


A collection of interconnected networks.

חיבור בין רשתות – Connecting Networks

- **בשכבה 1 (הפיזית) -** באמצעות משחזר (repeater) אשר מעתיק סיביות מרשת אחת לרשת אחרת, תוך הגברת האותות. הם אינם מבינים מאומה בפרוטוקולים אלא מחוללים מחדש את האותות.
- **בשכבה 2 (הערוץ) –** גשרים (bridges) ומתגים (switches). הגשר מקבל מסגרות, בודק את הכתובות שלהן, ומעביר אותן לרשת אחרת. יכול לשנות סיביות כותרת השייכות לנתוני בקרה של שכבת הערוץ.
- **בשכבה 3 (הרשת) –** נתבים מרובי פרוטוקולים (multiprotocol router) שולח מנות בין רשתות המשתמשות בפרוטוקולי שכבת רשת שונים.
- **בשכבה 4 (התובלה) –** שערי תובלה (transport gateway) מקשרים בין רשתות עם פרוטוקולי שכבת תובלה שונים.
- **בשכבה 5 (היישום) –** שער יישום (application gateway) המשמש לחיבור רשתות בשכבות הגבוהות.

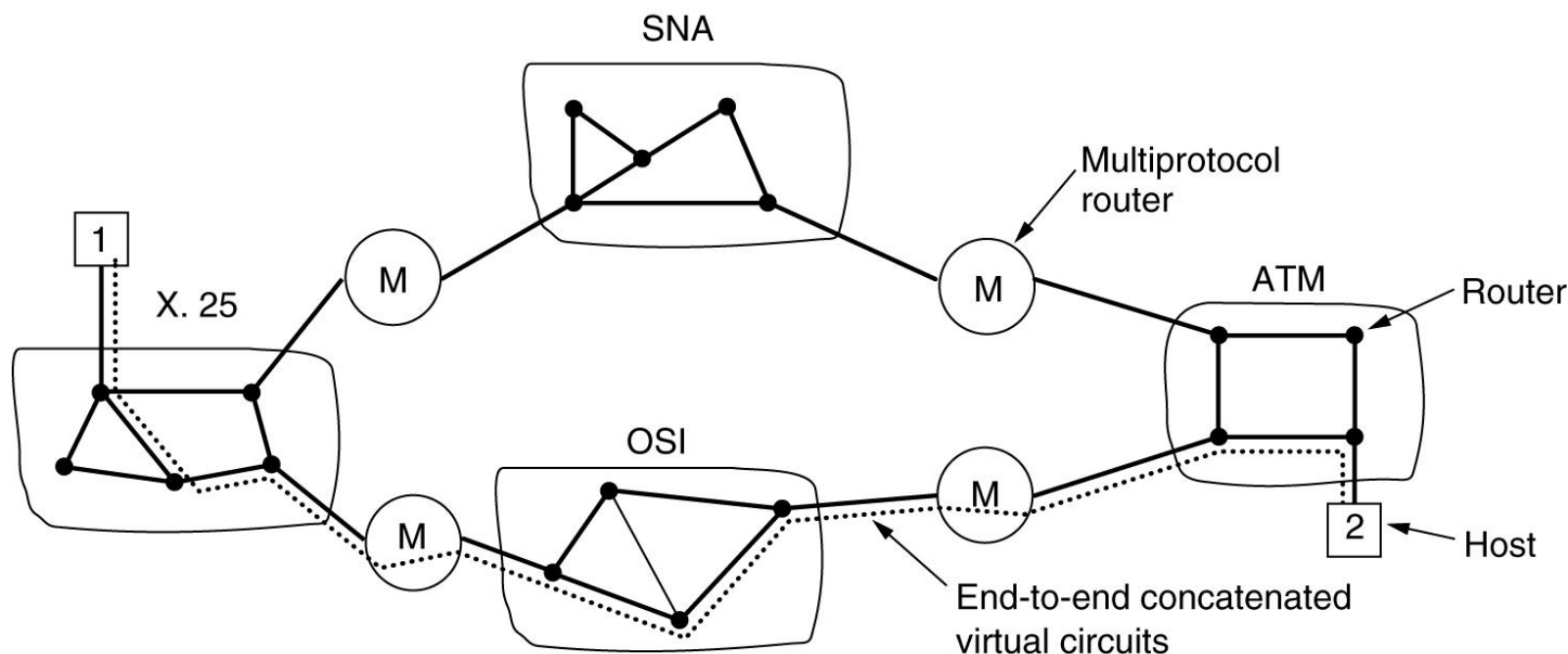
חיבור בין רשתות – Connecting Networks



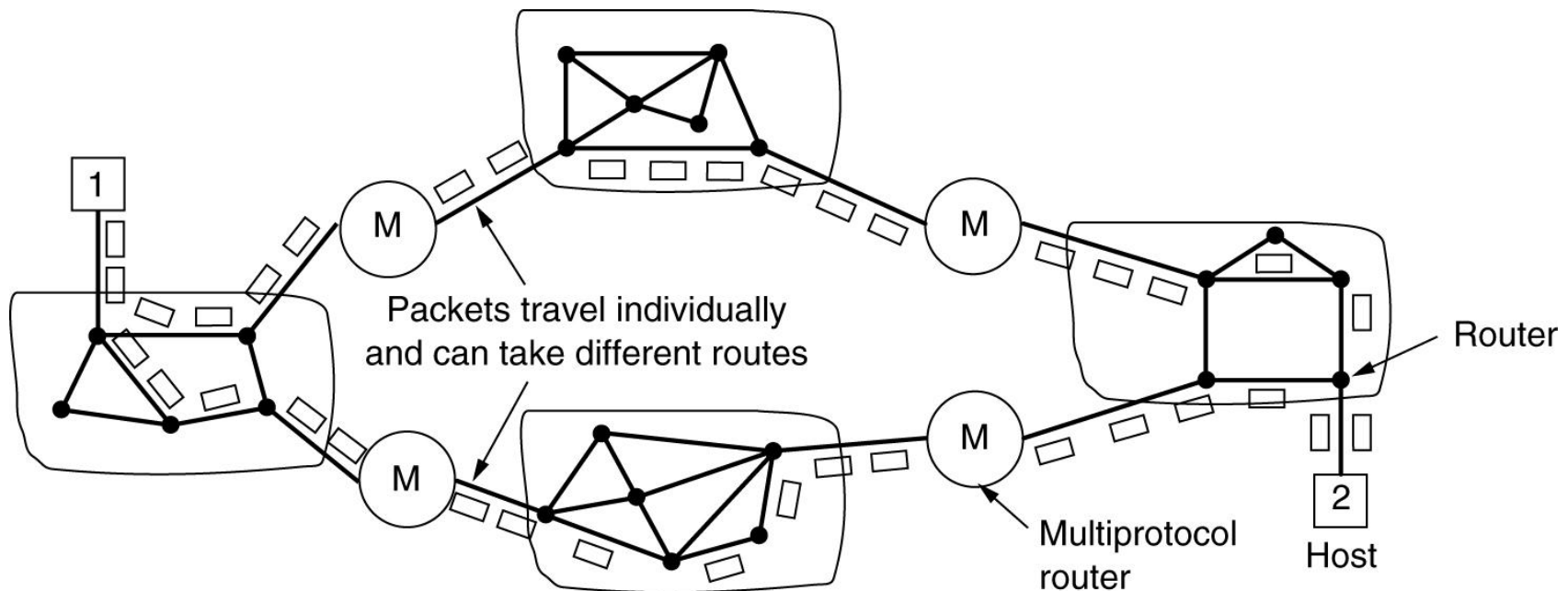
(a) רשתות אתרנט מחוברות ע"י מתג

(b) רשתות אתרנט מחוברות ע"י ראוטרים

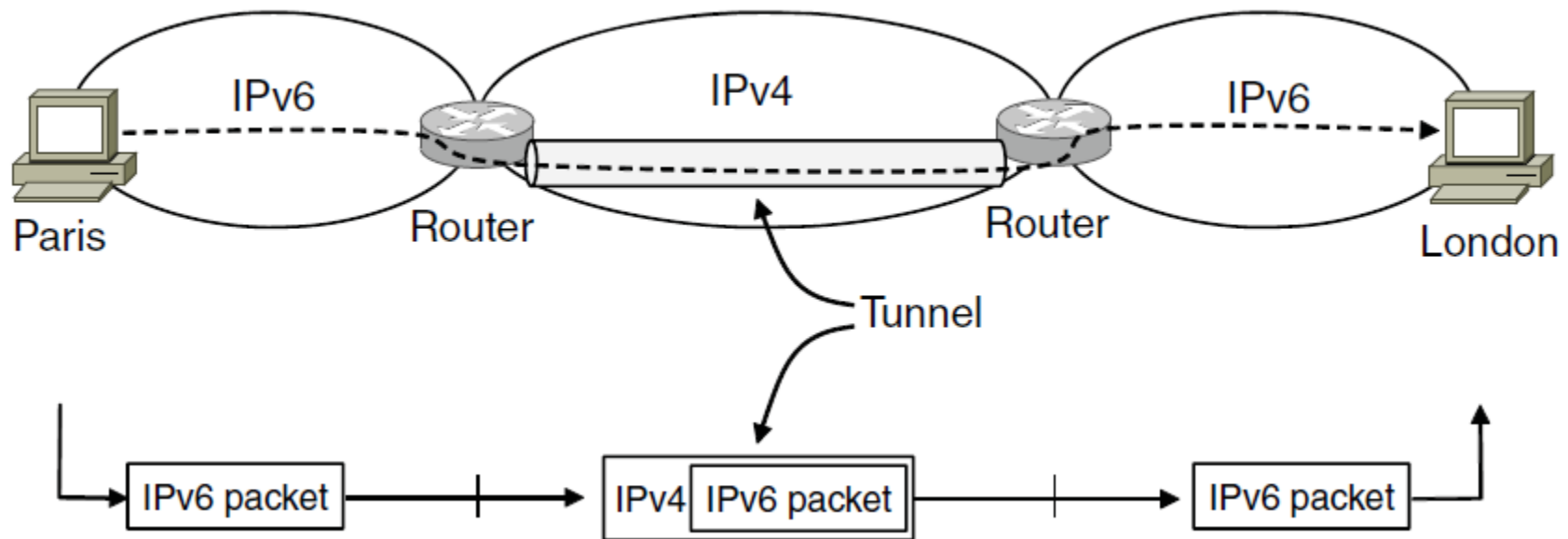
בשכבת הרשת יכול להיות חיבור מקושר הבנוי משרשר
של מעגלים מדומים - concatenated virtual circuits



חיבור לא מקושר - Connectionless Internetworking



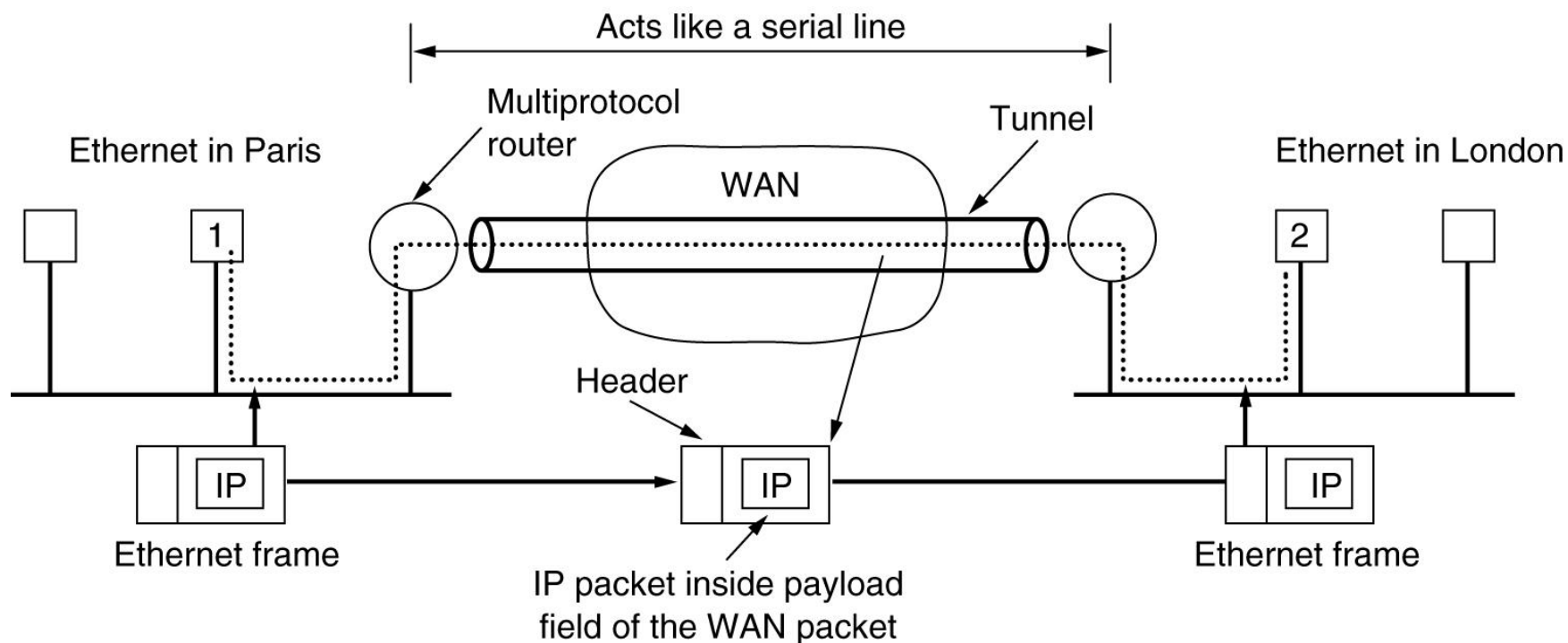
שימוש במנהרה - Tunneling



Tunneling a packet from Paris to London.

שקוף עבור רשתות המקור והיעד

שימוש במנהרה - Tunneling



Tunneling a packet from Paris to London.

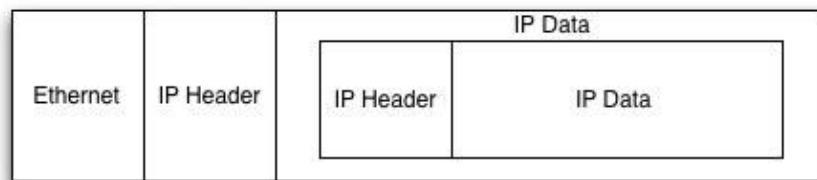
שקוף עבור רשתות המקור והיעד

שימוש במנהרה - Tunneling

Tunnels are a mechanism used to send unsupported protocols across diverse networks.

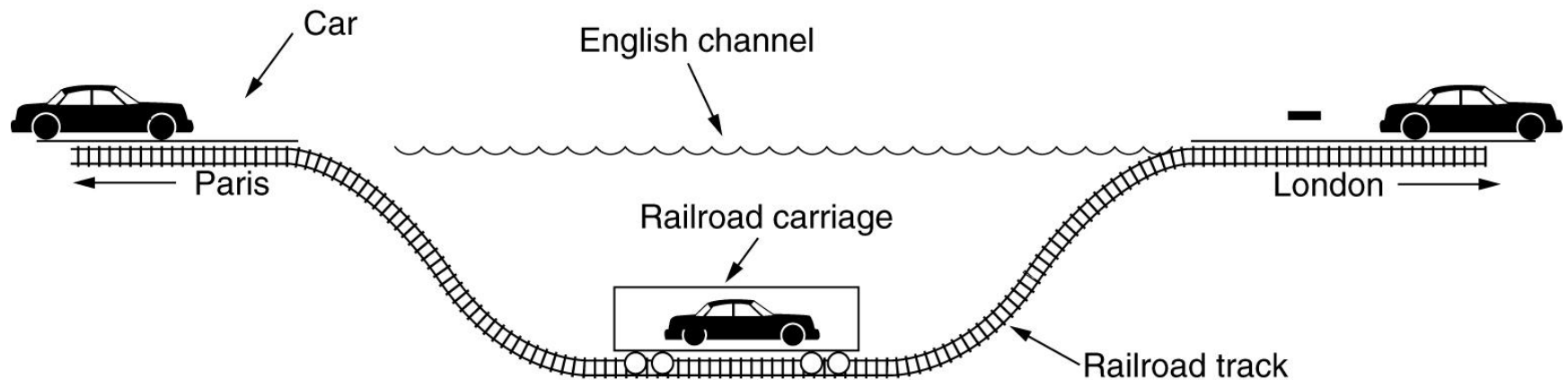
It is a technology that enables one network to send its data via another network's connections. Tunneling works by encapsulating a network protocol within packets carried by the second network.

Tunneled data adds to the size of the packet, resulting in less data being sent per-packet.



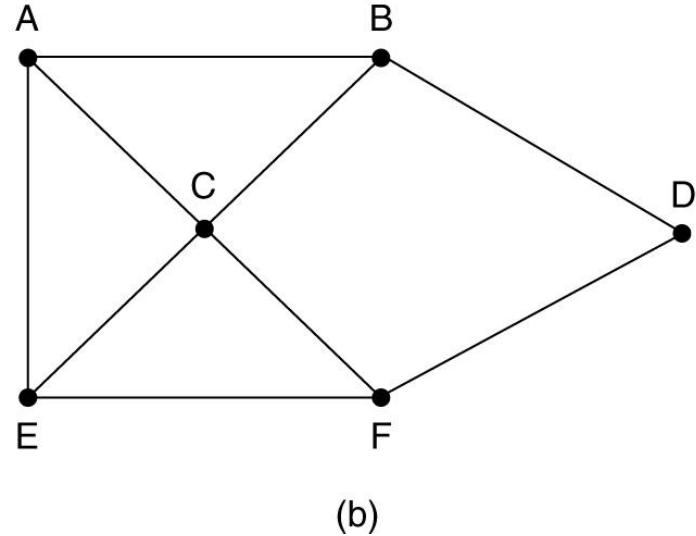
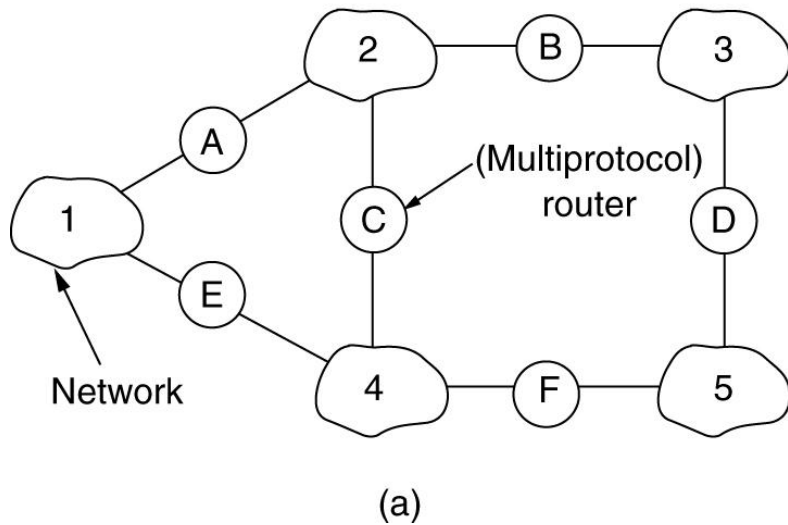
Tunneling is also called encapsulation.

שימוש במנהרה - Tunneling



Tunneling a car from France to England

ניתוב בין רשתות - Internetwork Routing



שני שלבים (כמו בניתוב מדרגי):

ניתוב פנים רשתי - Interior Gateway Protocol – IGP
שהוא ניתוב בתוך תחומי הרשת.

ניתוב בין רשתי - Exterior Gateway Protocol – EGP
שהוא ניתוב בין רשתות שונות.

ניתוב בין-רשתי –

התמודדות עם הגבלות שונות לגבי אורך המנות.

הפתרון - פיצול - Fragmentation

שתי גישות:

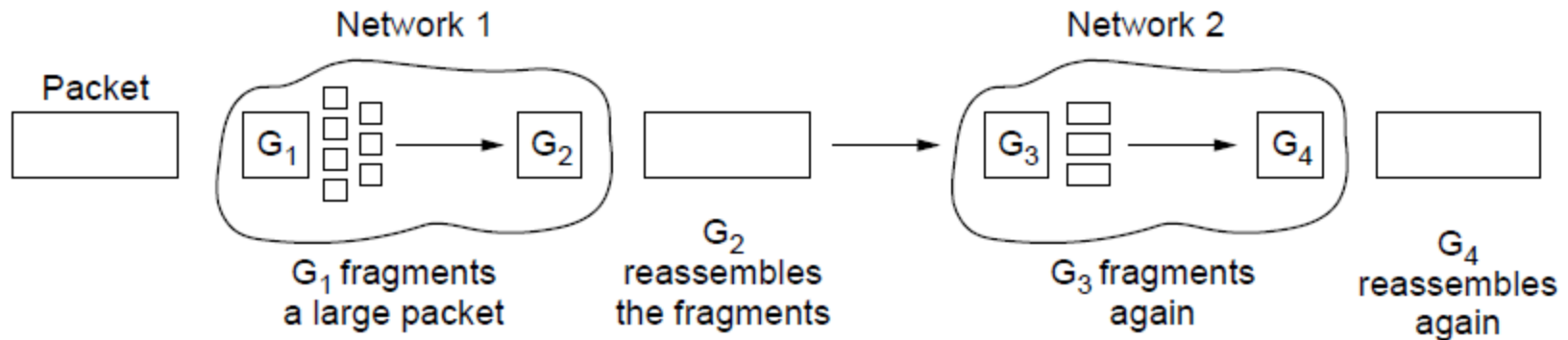
1. הפיצול והאיחוד נעשים בכל מעבר מרשת לרשת -

Transparent fragmentation

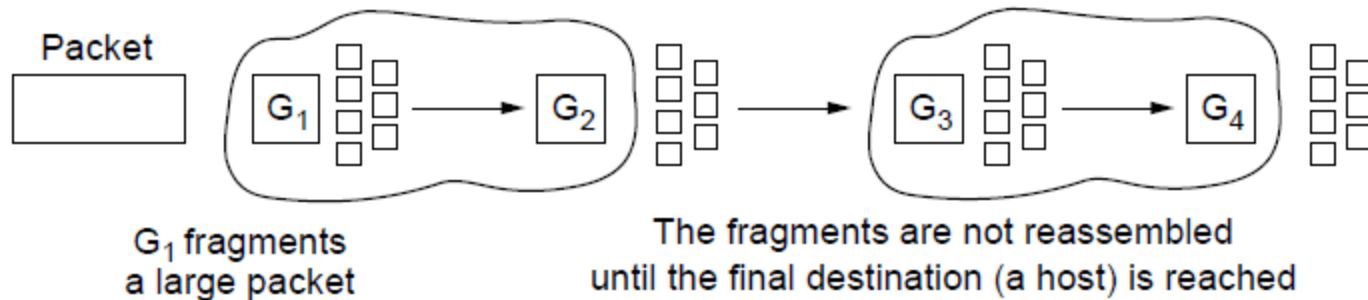
2. האיחוד מתבצע רק בשלב הסופי שבו המנה מגיעה

ליעדה – Nontransparent fragmentation

Fragmentation



(a)



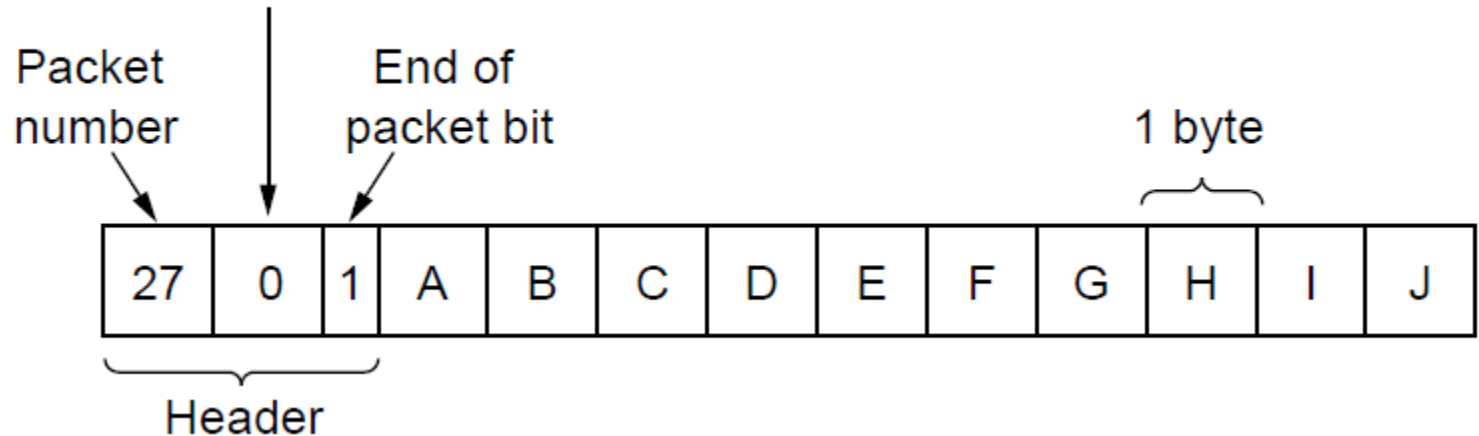
(b)

(a) – transparent fragmentation

(b) – nontransparent fragmentation

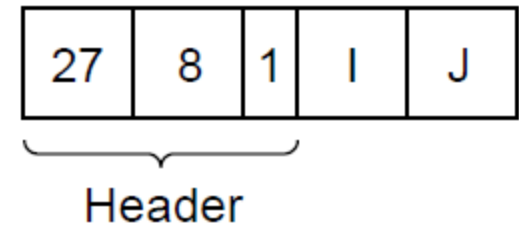
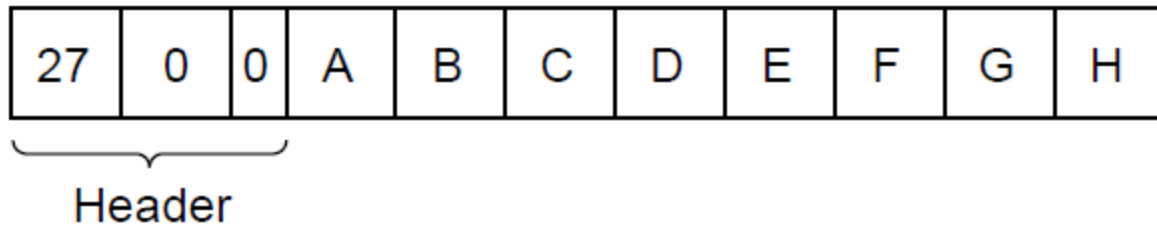
Fragmentation - דוגמא

Number of the first elementary fragment in this packet



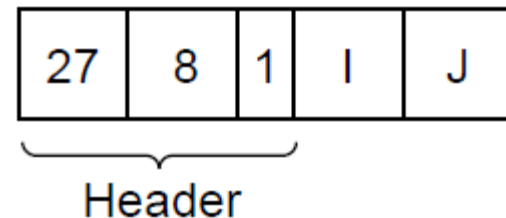
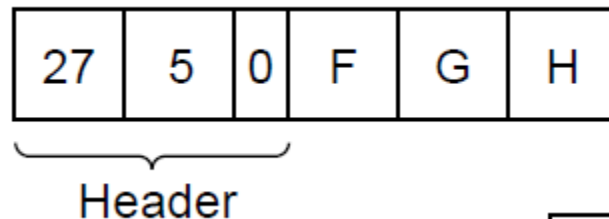
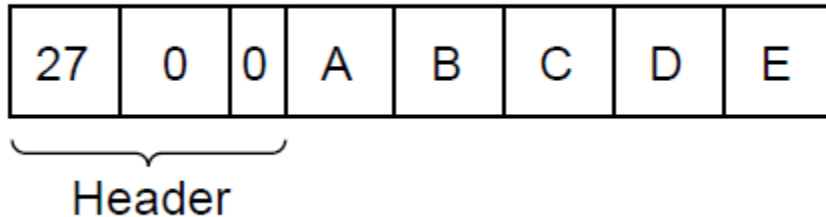
המנה המקורית, שמכילה 10 בתי דטה.

Fragmentation – דוגמא - המשך



חלוקת המנה במעבר בתת-רשת שבה הגודל המקסימלי של דטה במנה הוא 8 בתיים.

Fragmentation – דוגמא - המשך

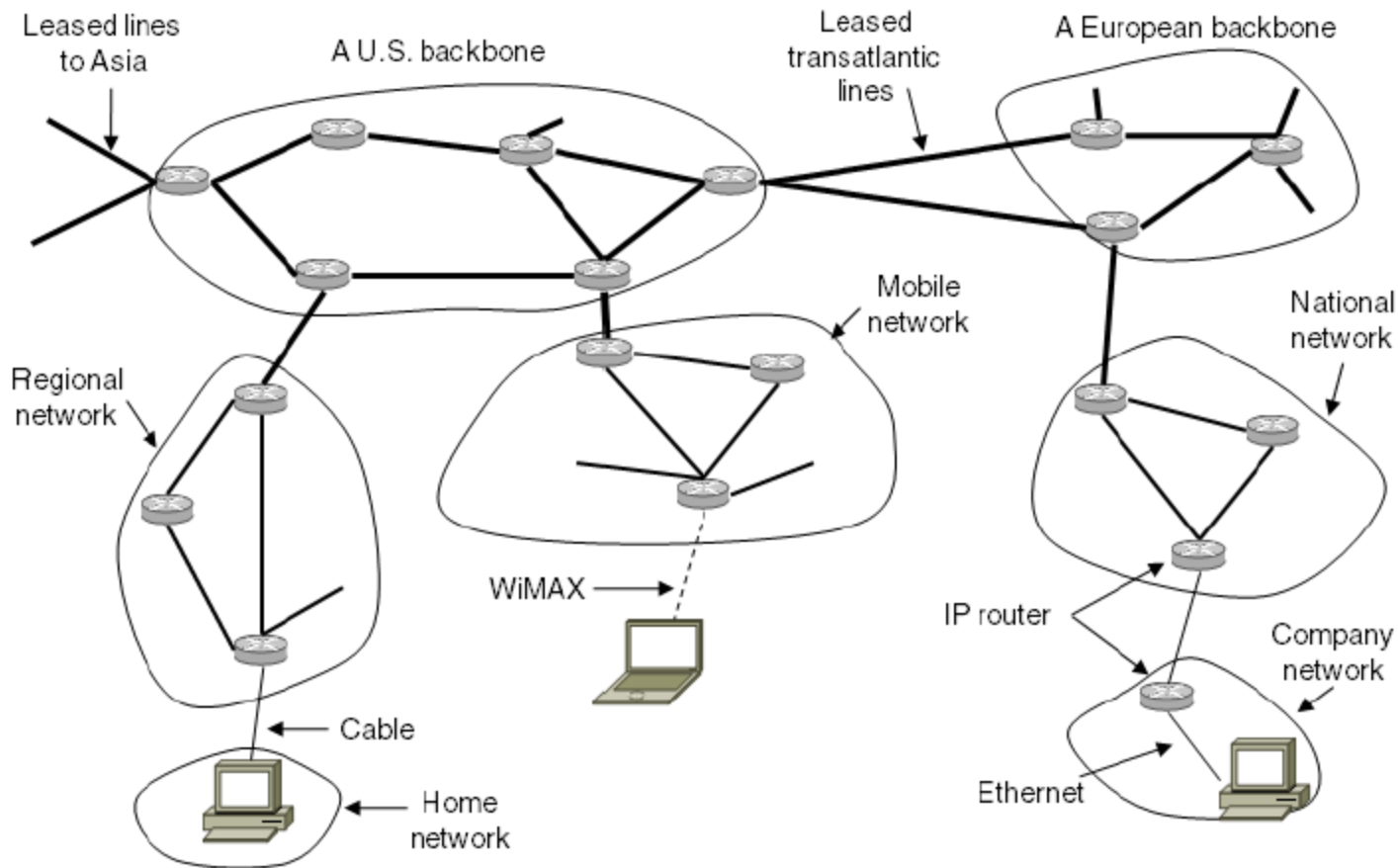


חלוקת המנה במעבר בתת-רשת שבה הגודל המקסימלי של דטה במנה הוא 5 בתיים.

שכבת הרשת באינטרנט

- פרוטוקול IP version 4
- כתובות IP
- פרוטוקול IP version 6

שכבת הרשת באינטרנט

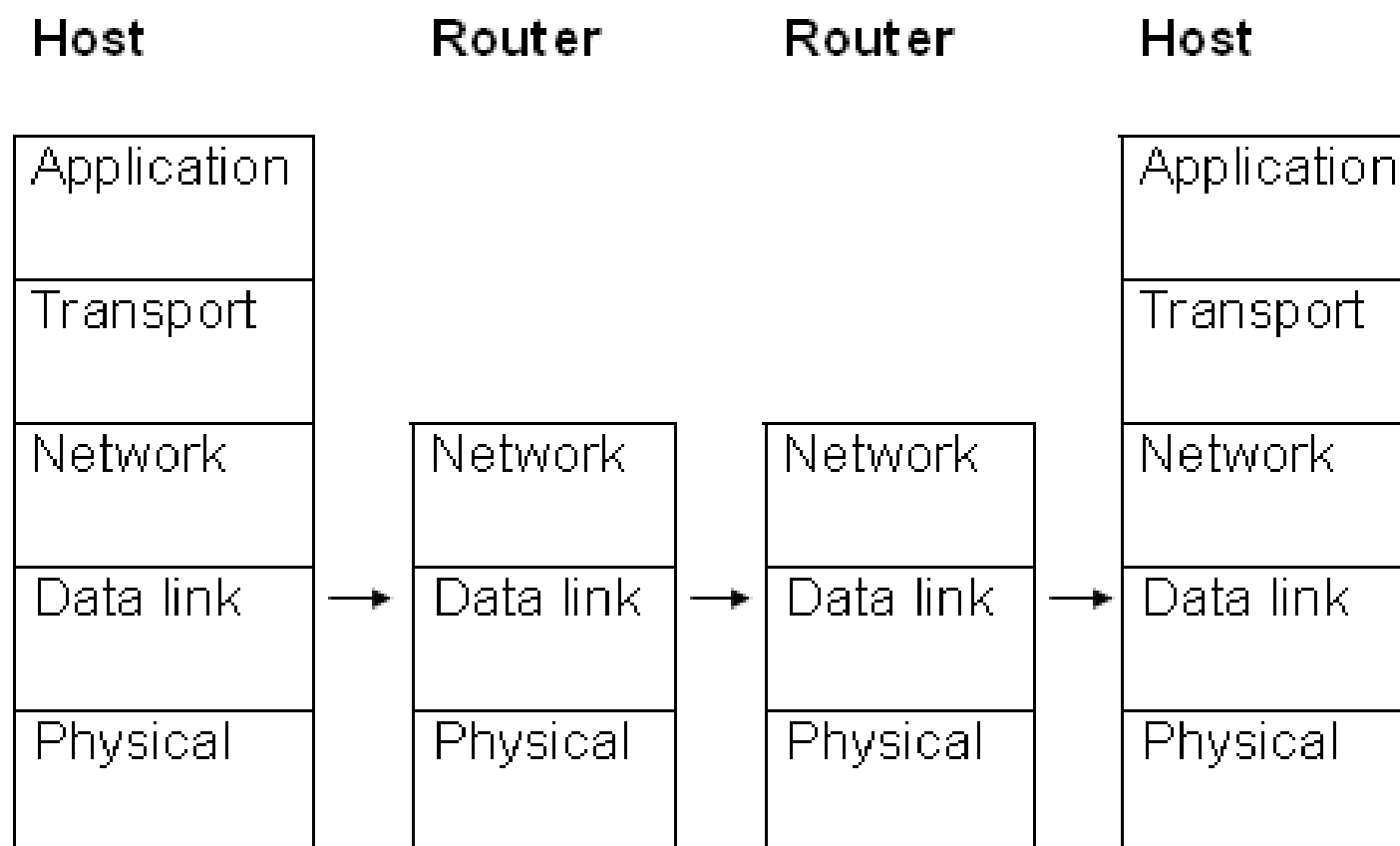


האינטרנט – אוסף של רשתות שמחוברות ביניהן

IP – פרוטוקול שכבת הרשת באינטרנט

- מגדיר את שיטת המיעון באינטרנט, את מבנה ההודעות ואת אופן הטיפול בהן.
- (אינו עוסק בנושאים כגון: ניתוב, טיפול בשגיאות ובקרת עומס).
- אינטרנט – רשתות שחוברו יחד. לא בהכרח מאותו סוג.
- במחשבים (HOSTS) קיימים כל השכבות וכל הפרוטוקולים.
- בנתבים (Routers) יש רק Network Core שזה: השכבה הפיזית, data link ו- IP.

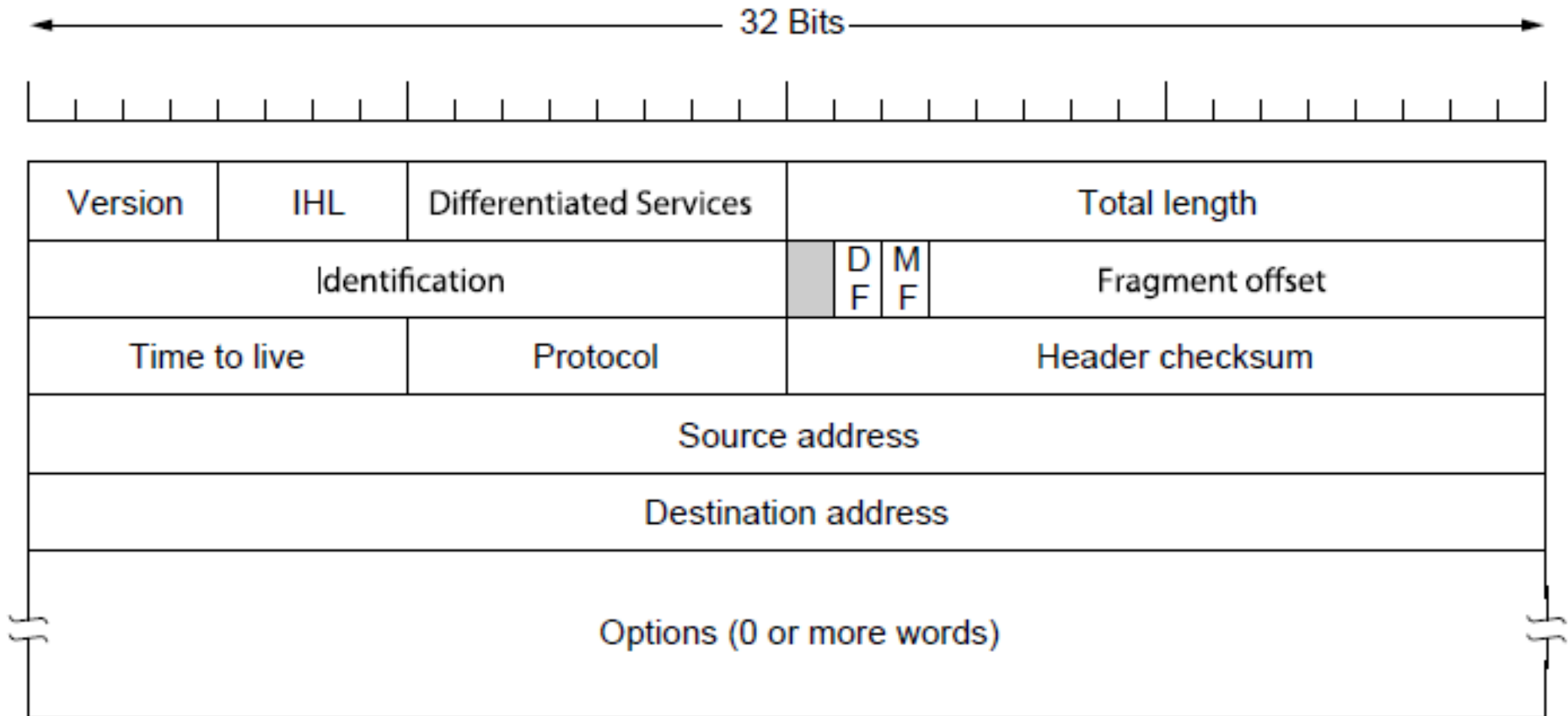
מבוא לרשתות תקשורת מחשבים – פרק 5



מטרתו של פרוטוקול IP:

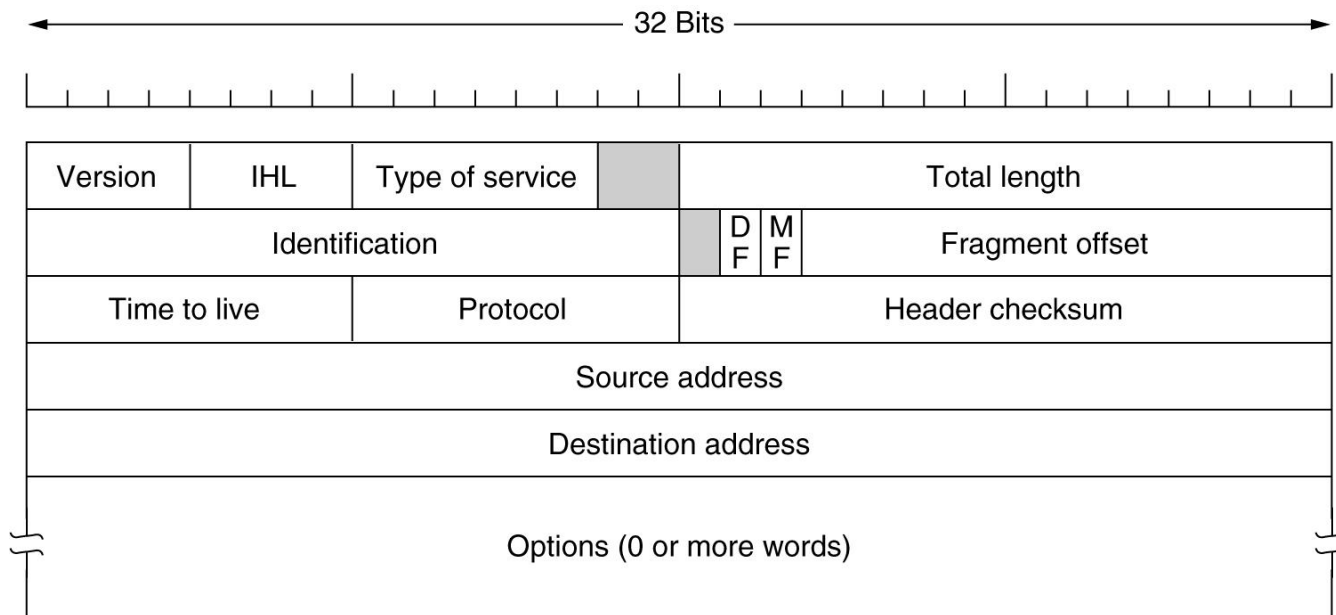
- לעשות את מירב המאמצים להעביר datagrams מהמקור ליעד, בידיעה שבדרך יש רשתות מסוגים שונים.
- מספק שרות לא מקושר ולא אמין.
- מוותרים על הקישור ועל האמינות בשביל המהירות והגמישות.
- שכבת התובלה מעבירה לשכבת הרשת datagrams שגודלן יכול להגיע עד 64KB, אבל בפועל הוא לא יהיה בדרך כלל יותר מ-1.5KB, בגלל מגבלת ה-ethernet.
- לעיתים החבילה תפוצל למספר חלקים לצורך העברתה.
- כאשר כל החלקים מגיעים לצד השני, הם מורכבים מחדש ומועברים לשכבת התובלה.

IP HEADER



The IPv4 (Internet Protocol) header.

מבוא לרשתות תקשורת מחשבים – פרק 5

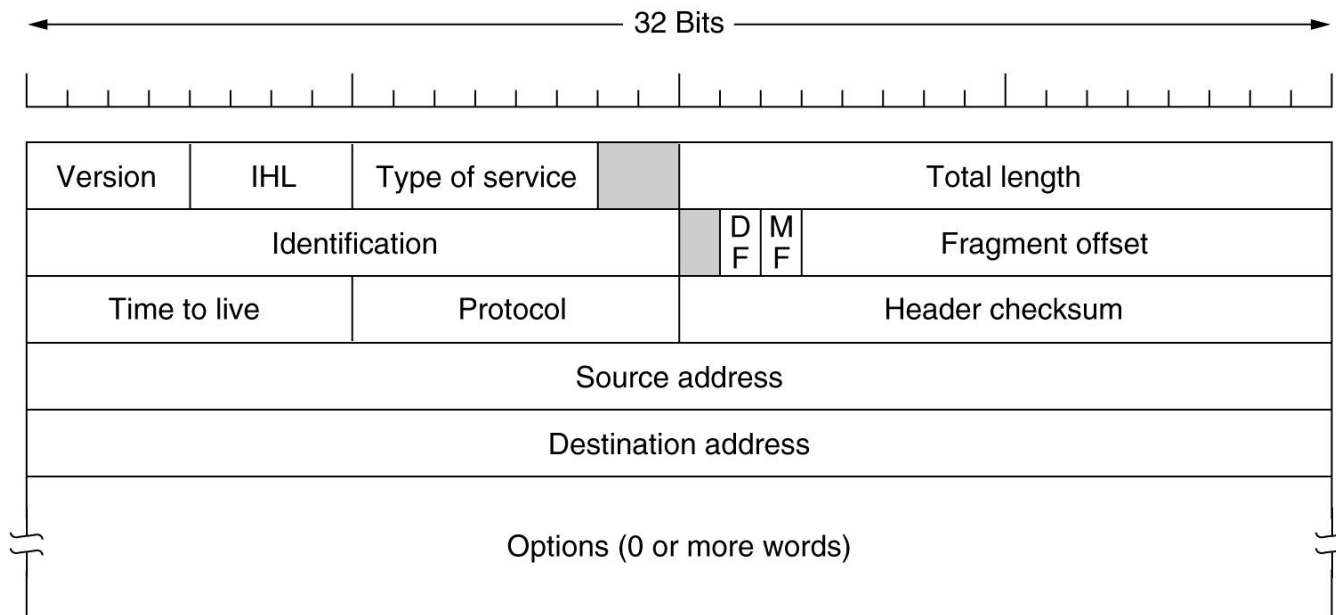


• **Version:** גרסת הפרוטוקול שבו משתמשים (הגרסה הנוכחית היא 4).

• **Header Length - IHL:** אורך הכותרת בכפולות של 32 סיביות (לדוגמא, כותרת בת 20 בתים תצוין באורך 5). אורך הכותרת לא יעלה על 60 בתים.

• **Type-Of-Service:** סוג השרות – למשל, האם להעדיף מהירות על פני אמינות.

מבוא לרשתות תקשורת מחשבים – פרק 5

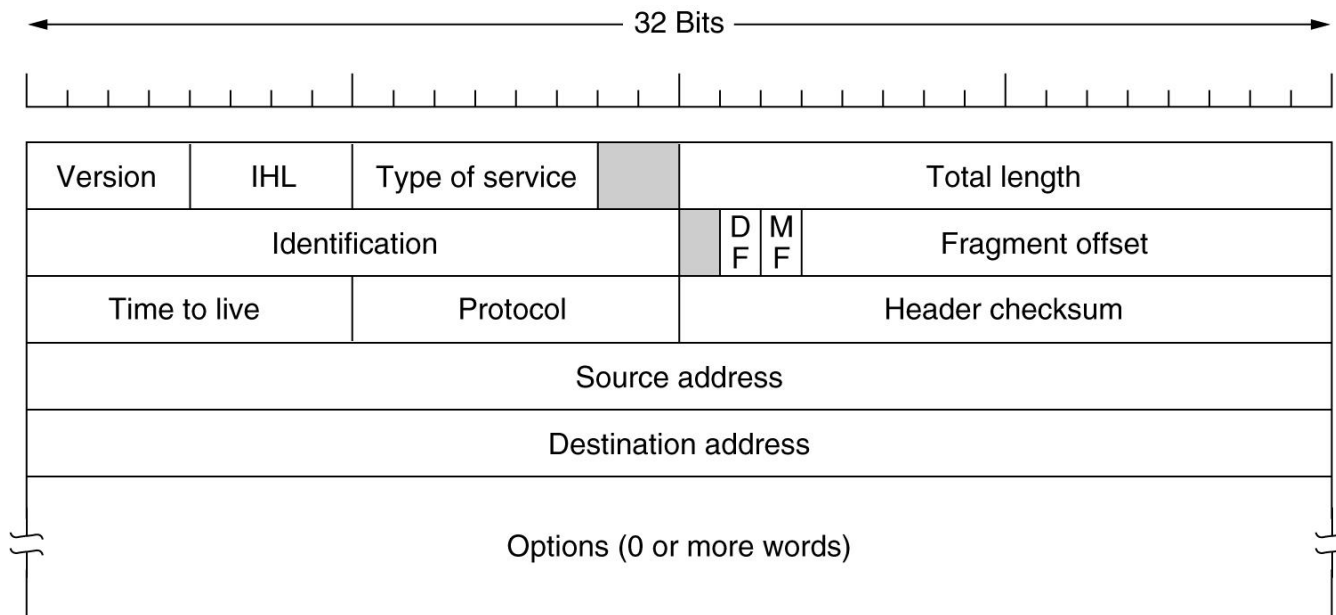


• **Total length:** גודל החבילה כולה (כותרת ונתונים) בביתים.

גודל מינימלי: 576 בתים, גודל מקסימלי: 65,535 בתים.

• **Identification:** מספר הזיהוי של החבילה. הזיהוי דרוש כאשר חבילה עוברת fragmentation, כדי שתחנת היעד תדע לזהות ולחבר מנות ששייכות לאותה חבילה (Datagram).

מבוא לרשתות תקשורת מחשבים – פרק 5

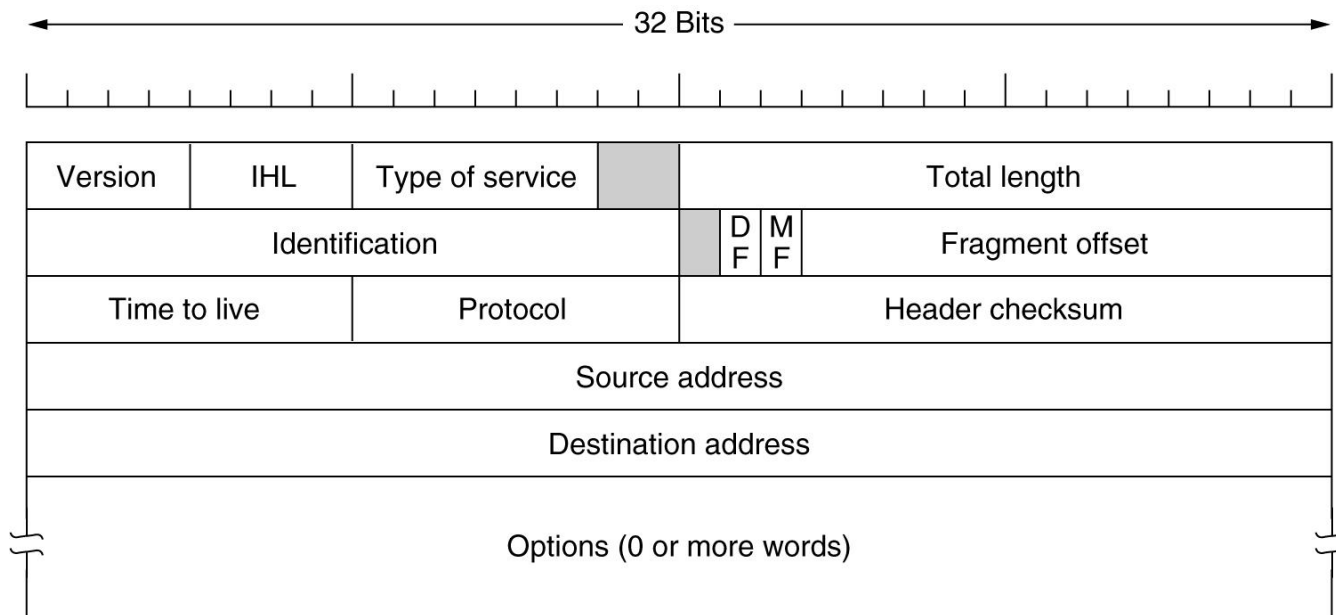


:Flags

הסיבית הראשונה שמורה לשימושים עתידיים, וערכה תמיד 0.

הסיבית השנייה **Don't Fragment – DF**, מציינת אם ניתן לחלק את הנתונים למקטעים (Fragmentation). =0 ניתן לחלק את רצף הנתונים לקטעים, =1 לא ניתן לחלק את הנתונים.

מבוא לרשתות תקשורת מחשבים – פרק 5

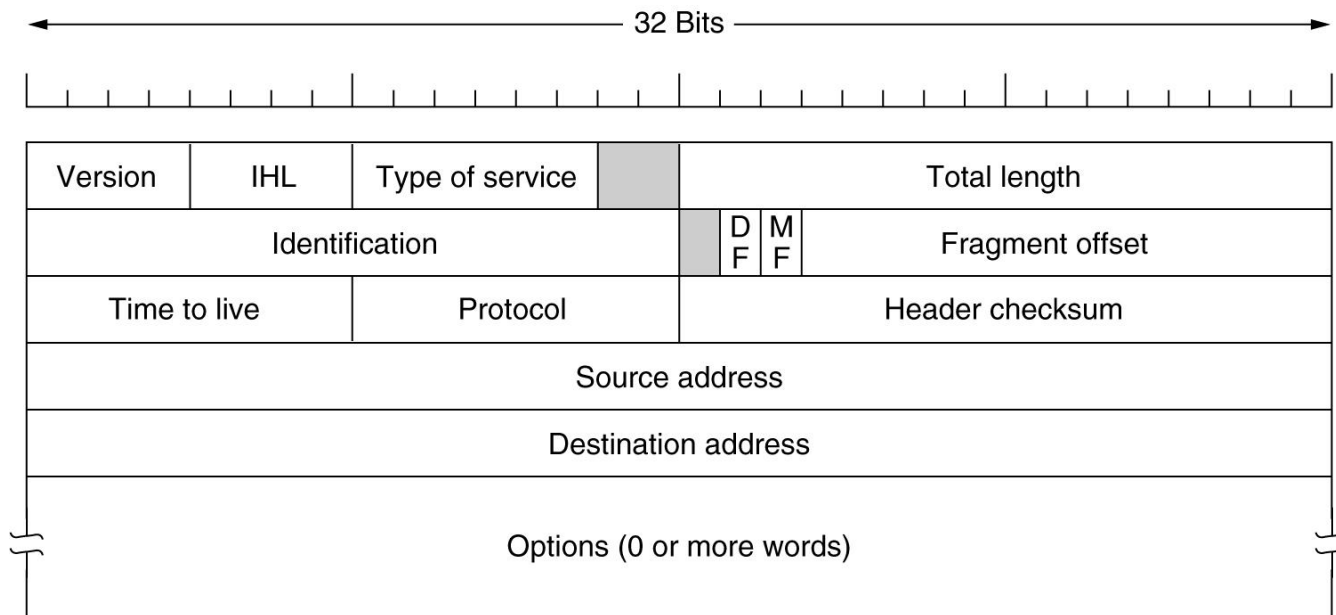


הסיבית השלישית **More Fragments – MF** מציינת אם המקטע (Segment) הוא האחרון בצרור הנתונים, או שיש לחכות לעוד מקטעים. $0 =$ החלק האחרון מכלל חלקי הצרור (או שהוא היחיד), $1 =$ יש קטעים נוספים השייכים לצרור נתונים זה.

Fragment Offset: המיקום היחסי של המקטע בתוך החבילה המקורית.

במקטע הראשון יופיע 0 , באחרים יופיע ההיסט בכפולות של 8 בתיים. דבר זה עוזר לתחנת היעד לבנות מחדש את ההודעה, לפי הסדר הנכון של המקטעים.

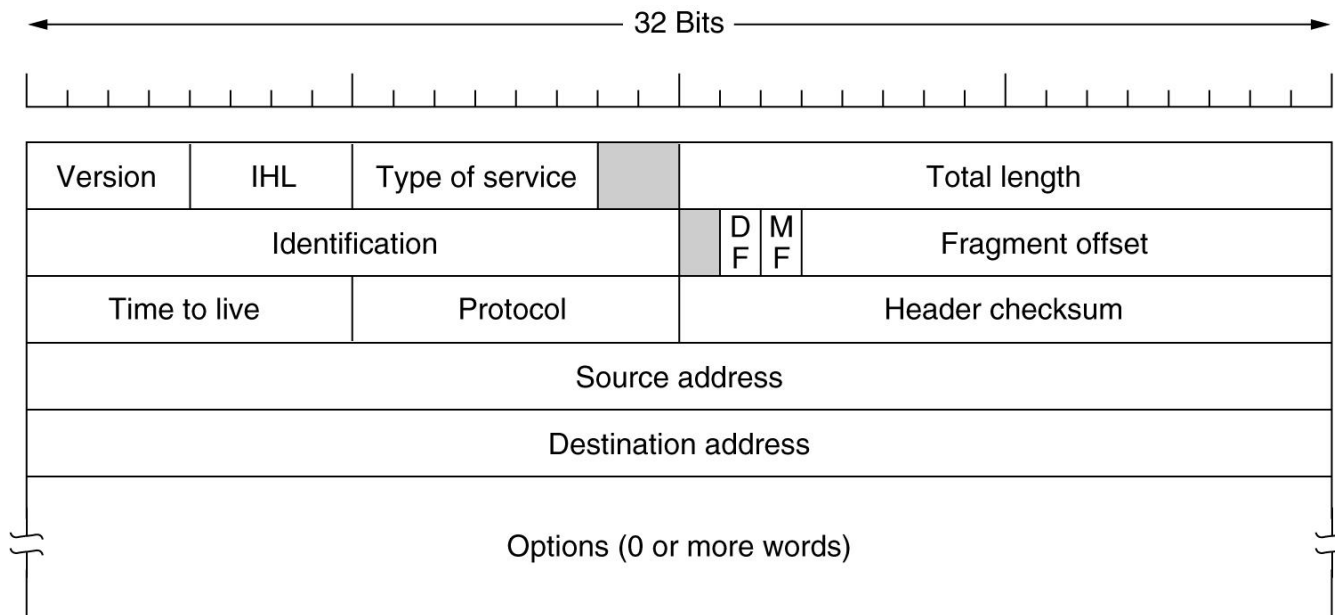
מבוא לרשתות תקשורת מחשבים – פרק 5



Fragment Offset

- כל המקטעים למעט האחרון, חייבים להיות בכפולות של 8 בתים, שזו יחידת המקטע הבסיסית.
- מכיון שלסימון ההיסט מוקצבות 13 סיביות, יכולים להיות מקסימום 8192 מקטעים בכל חבילה, כך שגודלה המקסימלי של חבילה יכול להיות 65,536 בתים.

מבוא לרשתות תקשורת מחשבים – פרק 5

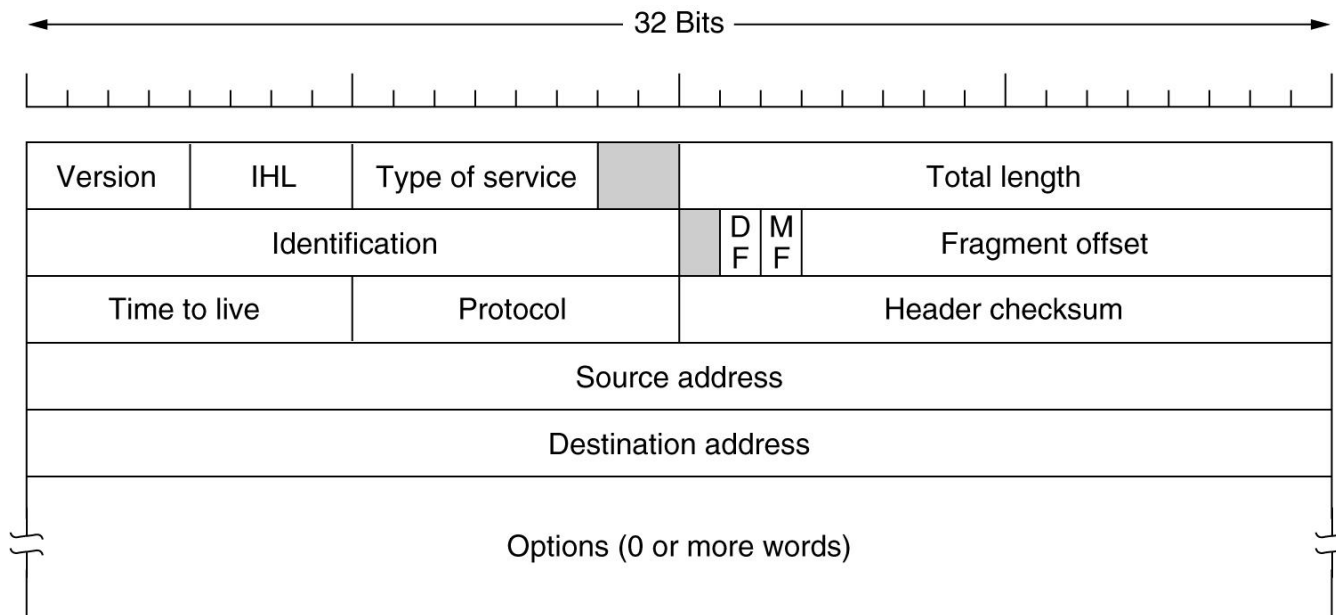


- **Time To Live (TTL):** מונה שערכו יורד בכל מעבר בנתב עד ל-0. תפקיד המנגנון הוא למנוע מהמנה לנוע בלולאה אין סופית ברשת.

- **Protocol:** כינוי הפרוטוקול בשכבה הרביעית, הגבוהה יותר, שימשיך לטפל במנת הנתונים, לדוגמא, TCP, UDP וכו'.

- **Header Checksum:** שדה העוזר להבטיח אמינות הכותרת.

מבוא לרשתות תקשורת מחשבים – פרק 5



- **Source Address**: כתובת ה-IP של תחנת המקור.
- **Destination Address**: כתובת ה-IP של תחנת היעד.
- **Options**: סט של אופציות שניתן להוסיף לחבילה, כמו דרישה למסלול מסוים. אורך רשימת האופציות עד 40 בתים.
- **Data**: שדה המכיל מידע משכבות עליונות.

כמה מהאופציות ב-IP header

| Option | Description |
|-----------------------|--|
| Security | Specifies how secret the datagram is |
| Strict source routing | Gives the complete path to be followed |
| Loose source routing | Gives a list of routers not to be missed |
| Record route | Makes each router append its IP address |
| Timestamp | Makes each router append its address and timestamp |

Internet Checksum

- מחלקים את הכותרת ליחידות בנות 16 סיביות כ"א. אם יש צורך מוסיפים אפסים ליחידה האחרונה.
- סוכמים את כל היחידות בשיטת המשלים ל-1. אם יש שארית, מוסיפים אותה לחיבור.
- בסוף הפעולה לוקחים את המשלים ל-1 של התוצאה, כלומר היפוך של הסיביות.
- אותה שיטה משמשת גם את TCP ואת UDP, ולכן נקראת Internet Checksum.
- ה-Checksum מחושב רק על הכותרת (IP Header) ולא על הנתונים.
- מחושב כל פעם מחדש כי שדה TTL משתנה בכל מעבר.

דוגמא:

Internet Checksum

1011

1000

שתי המילים

1000

שלב א': סוכמים

1011

1 0011

שלב ב': את השארית מחברים שוב 1

0100

שלב ג': לוקחים את המשלים ל-1 1011

לתשומת לב: את ה-checksum מחשבים רק על ה-header לא על התוכן!

כתובת IP

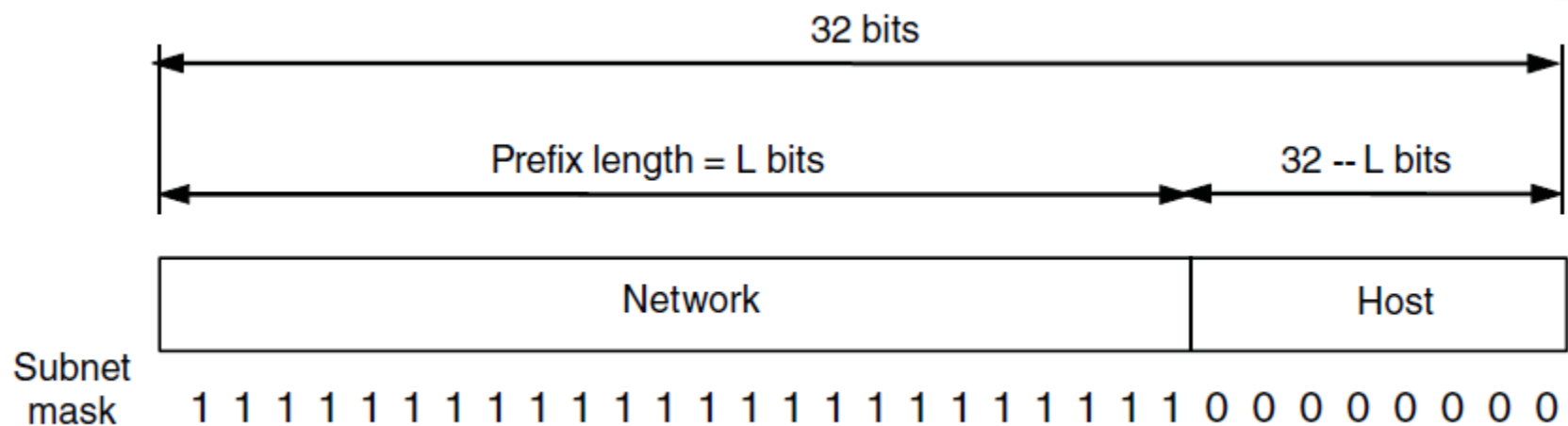
מורכבת מ-32 סיביות בקבוצות של 8 מופרדות בנקודה,
או 4 מספרים בין 0 ל-255 כ"א.

192.41.6.20

לדוגמא: הכתובת:

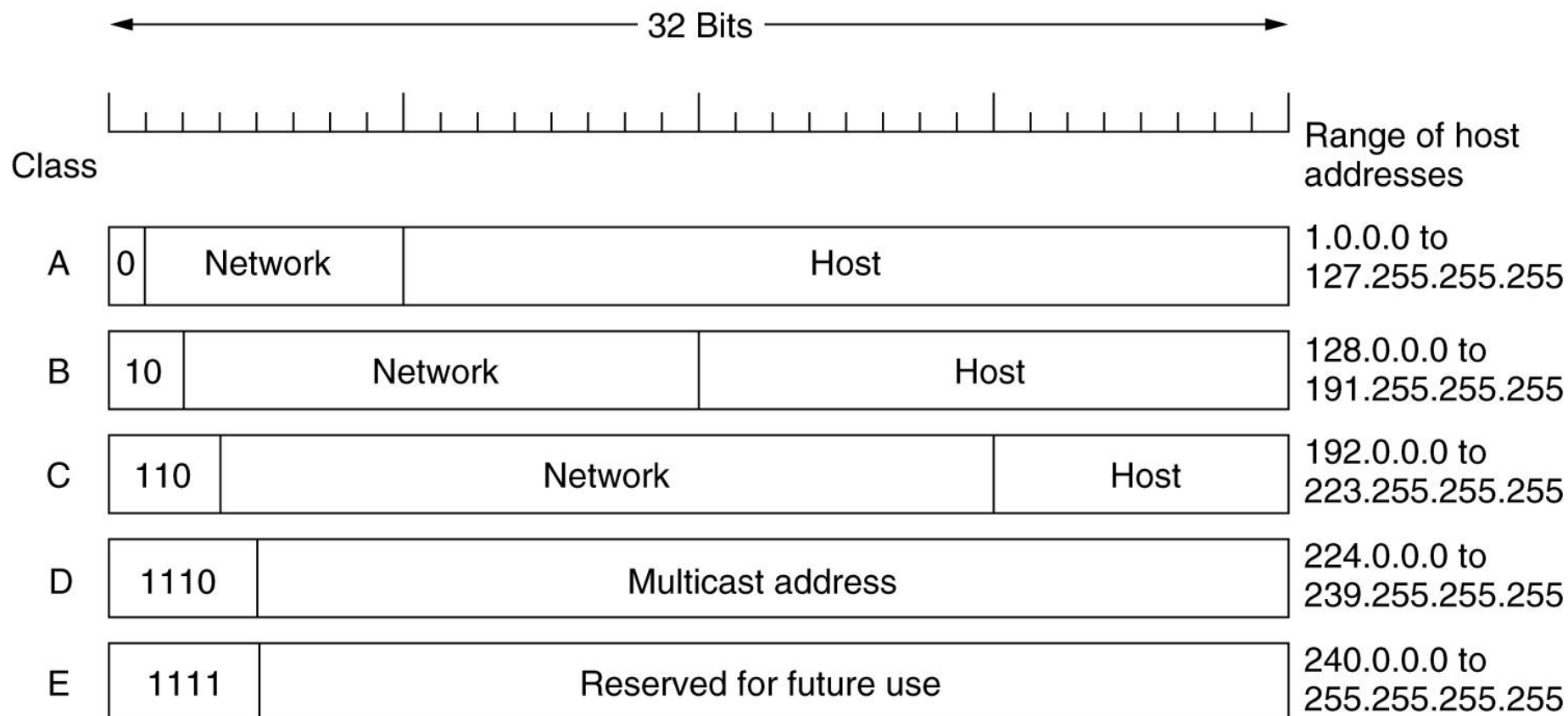
שקולה ל-

11000000.00101001.00000110.0010100



Classful Addressing – כתובות

מרחב הכתובות מחולק ל-5 מחלקות, לפי גודל הרשתות הדרושות:



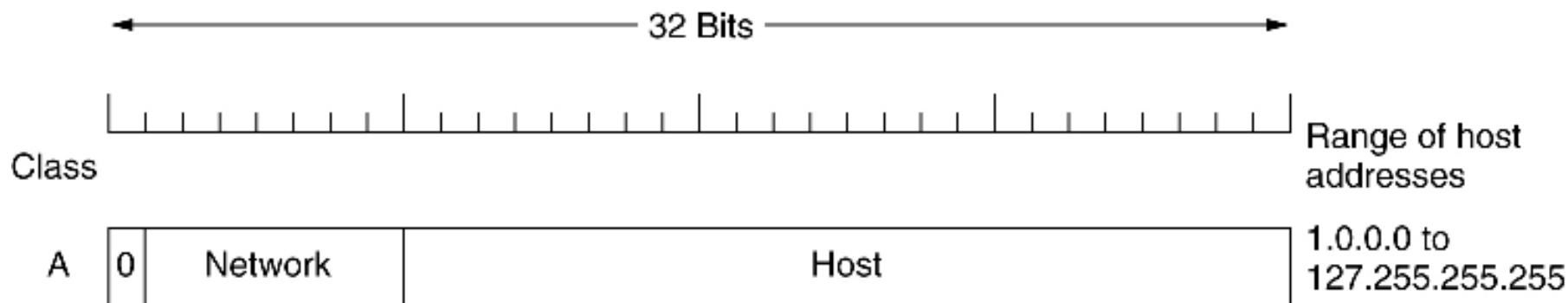
מבוא לרשתות תקשורת מחשבים – פרק 5

אין אף מחלקה המהווה רישא (prefix) של סיביות המחלקה האחרת.

ברגע שמזהים את סדרת הסיביות המגדירה מחלקה מסוימת בתחילת הכתובת, יודעים בוודאות כי הכתובת שייכת למחלקה זו. תכונה זו נקראת **no-prefix**.

לדוגמה, ברגע שמזהים את הסיבית 1, בודקים את הסיבית הבאה. אם אחריה מופיעה הסיבית 0 – ברור שהמחלקה היא B (01) ואין צורך להמשיך ולבדוק. אם לאחריה מופיעה הסיבית 1, יש צורך להמשיך לבדוק את שאר הסיביות, כי ייתכן שקוד המחלקה הוא 110 או 1110 או 1111.

• מחלקה A

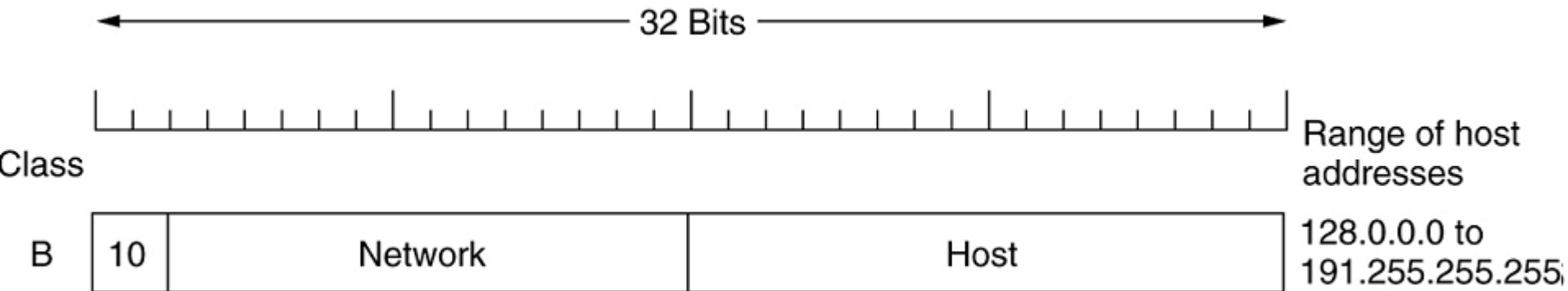


• מתחילה תמיד ב-0. כתובת הרשת מיוצגת על ידי 7

סיביות. מאפשרת 126 $(2^7 - 2)$ רשתות שונות

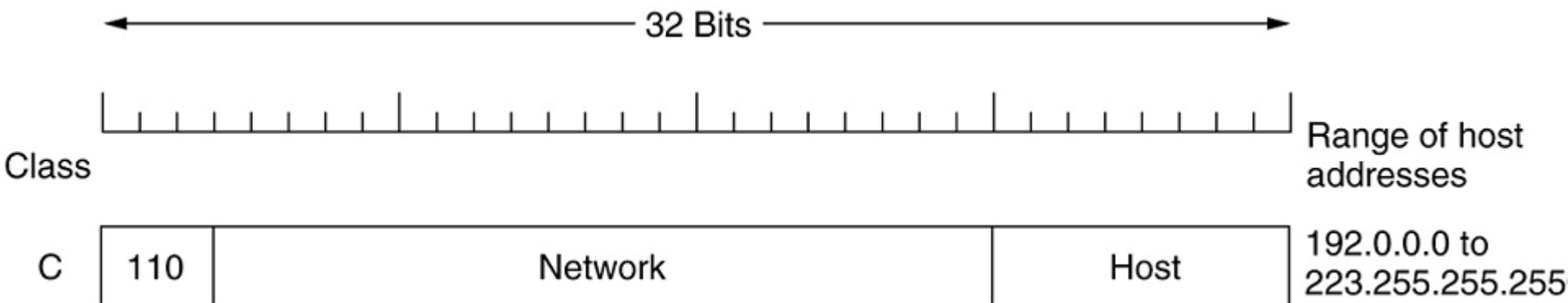
• $(1-126)$, עם עד כ-16 מיליון תחנות בכל רשת (2^{24}) .

• מחלקה B



- מתחילה ב-10 שאחריהן מספר רשת בן 14 סיביות, ומספר מחשב בן 16 סיביות. מאפשרת 2^{14} 16,384 רשתות שונות (128-191), עם עד כ- 65,534 תחנות בכל רשת.

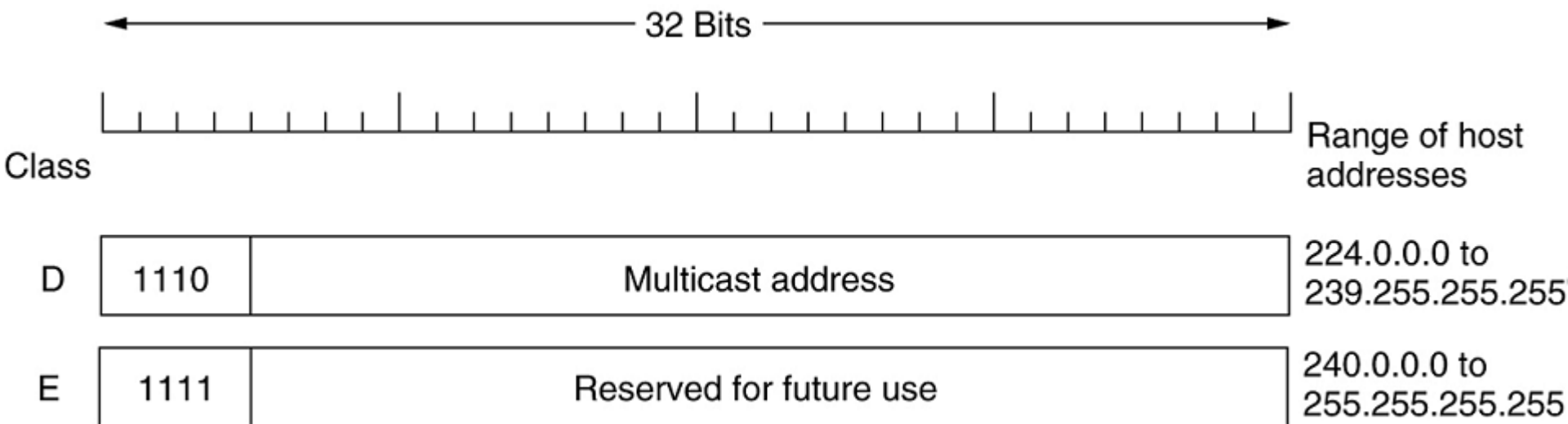
• מחלקה C



- מתחילה ב-110 ואחריהן כתובת רשת בת 21 סיביות, וכתובת מחשב בת 8 סיביות. מאפשרת 2,097,152 רשתות עם עד 254 תחנות בכל רשת.

מבוא לרשתות תקשורת מחשבים – פרק 5

- **מחלקה D** – מתחילה ב-1110 מיועדת לשידורים מרובי יעדים. 22 סיביות מוקצות להגדרת כתובות של קבוצת יעדים.
- **מחלקה E** – מתחילה ב-1111 ושמורה לשימוש עתידי.



כתובות IP מיוחדות

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|--|--|--|--|------------|--|--|--|--|-----|--|--|--|--|---------|--|--|--|--|-----|--|--|--|--|-----|--|--|--|--|--------------------------------|--|--|--|--|---------|--|--|--|--|------------------------|--|--|--|--|--------------------------------|
| 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | This host | | | | | | | | | | | | | | | |
| 0 0 | | | | | | | | | | ... | | | | | | | | | | 0 0 | | | | | | | | | | Host | | | | | | | | | | A host on this network | | | | | |
| 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | Broadcast on the local network | | | | | | | | | | | | | | | |
| Network | | | | | | | | | | | | | | | 1 1 1 1 | | | | | | | | | | ... | | | | | | | | | | 1 1 1 1 | | | | | | | | | | Broadcast on a distant network |
| 127 | | | | | (Anything) | | | | | | | | | | | | | | | | | | | | | | | | | Loopback | | | | | | | | | | | | | | | |

חלוקה פנים-ארגונית לתת-רשתות

• קבלת כתובת רשת חדשה היא בעייתית עקב המחסור ההולך וגובר בכתובות ה-IP.

• כלפי חוץ כל המחשבים בארגון מחוברים לרשת מקומית אחת, כאשר נתב מרכזי אחד, שמחובר לספק התקשורת ודרכו לאינטרנט, מחבר בין כל הרשתות המקומיות.

• כל המנות שמיועדות לאחד המחשבים בארגון, מגיעות לנתב המרכזי, אשר אחראי לשליחת כל מנה לרשת המקומית אליה שייך מחשב היעד.

תת-רשתות - subnets

• אם מדובר בארגון בעל מחשבים רבים, יידרש זמן מחשב יקר לסריקת כל הטבלה עבור כל מנה שמגיעה.

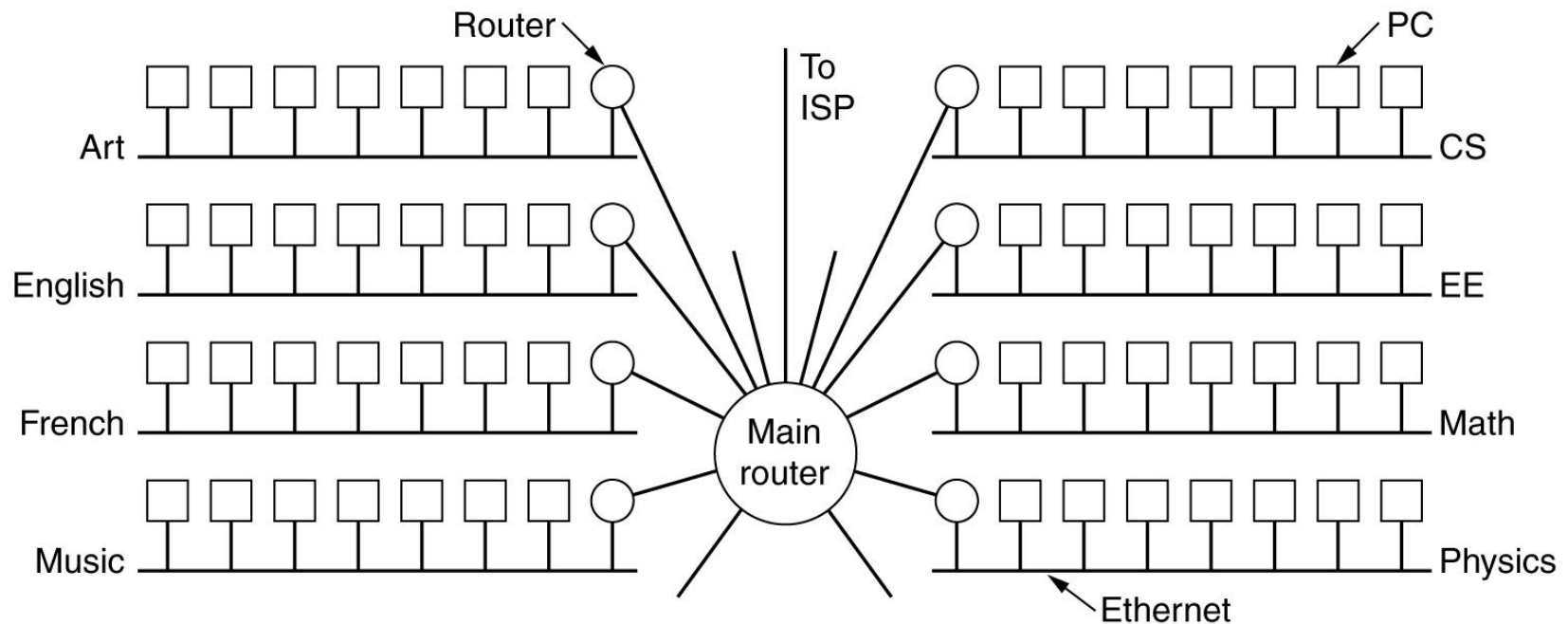
• כדי לייעל את התהליך, מקצים מספר סיביות משדה כתובת המחשב של כתובת ה-IP לטובת מספור הרשתות המקומיות שבתוך הארגון.

תת-רשתות - subnets

- כל נתב מחזיק מידע לגבי:
- איך להגיע לכל מחשב ב-subnet שלו.
- איך להגיע לכל subnet אחרת ברשת.
- אינו מחזיק מידע לגבי מחשבים אחרים ב-subnets אחרות.

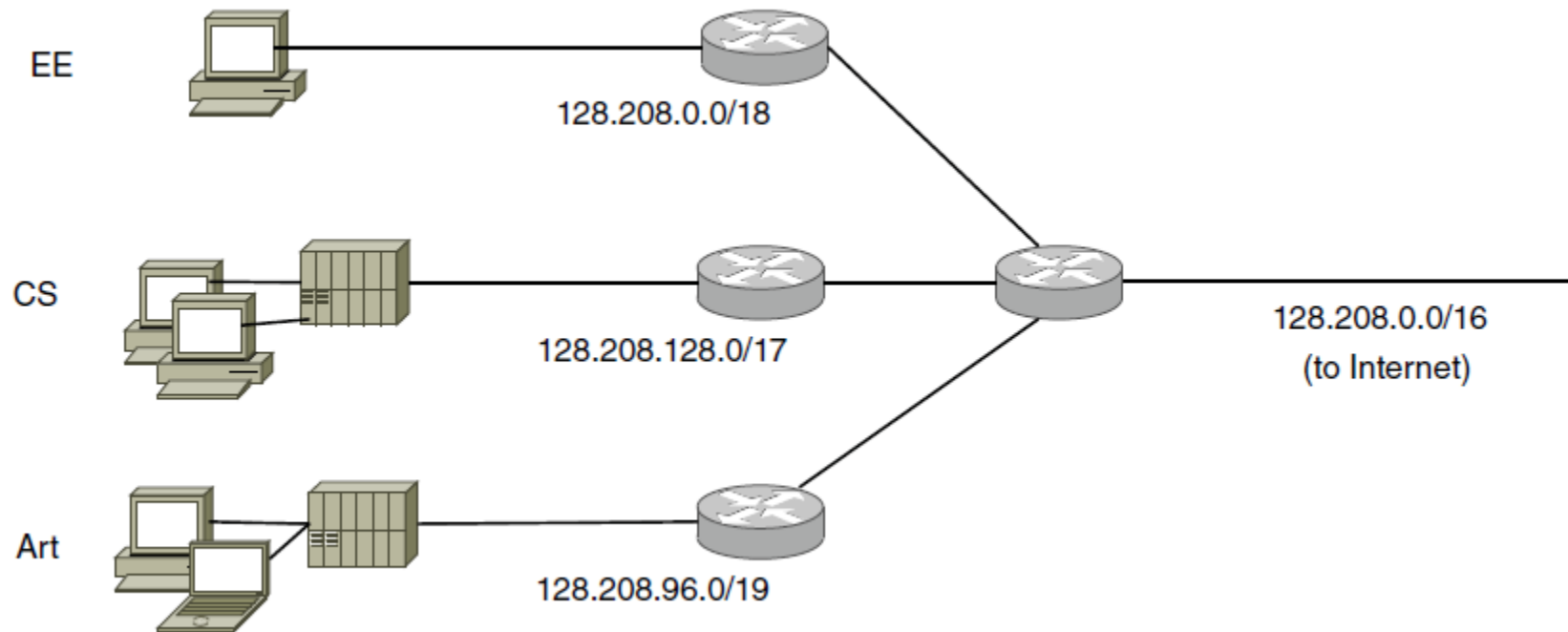
תת-רשתות Subnets

כלפי חוץ המחשבים בארגון מחוברים לרשת מקומית אחת.



רשת בקמפוס המבוססת על רשתות LAN של מחלקות שונות

כתובת IP



Splitting an IP prefix into separate networks with subnetting.

תת-רשתות Subnets

מחלקים את החלק של מזהה המחשב (host id) לשני חלקים: מזהה תת-הרשת (subnet id) ומזהה המחשב בתוך תת-הרשת. על מנת לאפשר ניתוב בין תת-רשתות, צריך לדעת איזה חלק של הכתובת מייצג את הרשת ואת תת הרשת, ואיזה חלק מייצג את המחשב. לשם כך, נעשה שימוש ב"מסכת תת-הרשת" (subnet mask) אשר מציינת כמה ביטים משמשים לזיהוי הרשת וכמה לזיהוי תת-הרשת.

אם היו Y ביטים עבור כתובת ה-host, והקצינו X ביטים מהם לטובת כתובת ה-subnet, יכולות להיות לנו 2^X תת-רשתות, ובכל אחת מהן עד $2^{(Y-X)}$ מחשבים.

תת-רשתות Subnets

דוגמא:

הרשת 131.12.0.0

(ביצוג בינרי – 10000011 00001100 00000000 00000000)

שייכת למחלקה B, לכן 16 הסיביות הראשונות שלה שייכות לרשת, ו-16 הסיביות האחרונות שייכות למשתמש.

מנהל הרשת יכול להגדיר שארבע סיביות נוספות שייכות לרשת, ובכך הוא מחלק את הרשת שלו ל-16 רשתות שונות, כאשר בכל רשת יהיו עד 4096 מחשבים.

מסכת הרשת תהיה עכשו 255.255.240.0, או בייצוג בינרי:

11111111 11111111 11110000 00000000

תת-רשתות Subnets

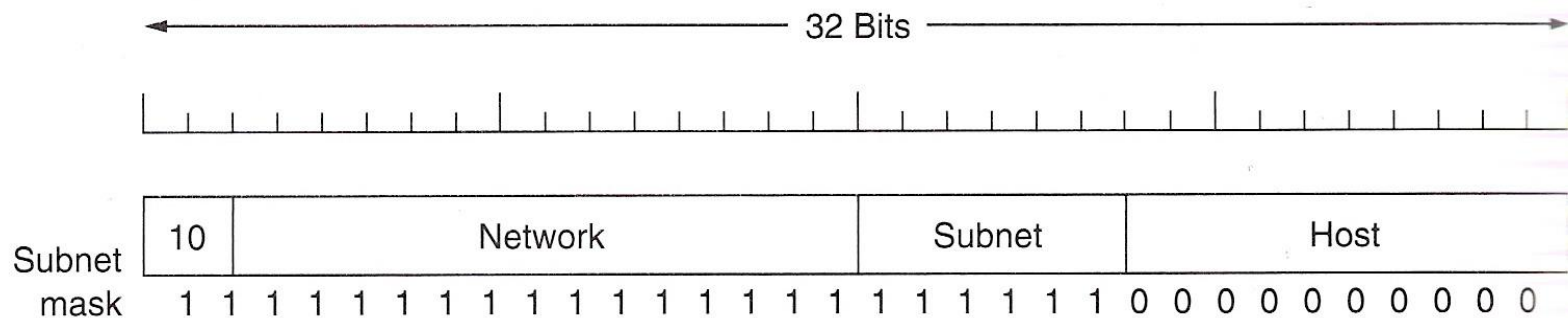


Figure 5-58. A class B network subnetted into 64 subnets.

הניתוב – באמצעות mask

ה-mask – מחרוזת סיביות הכוללת 1-ים עבור חלק הרשת והתת-רשת, ו-0-ים עבור חלק ה-host. מבצעים and על הכתובת ועל ה-mask, כך שחלק ה-host מתאפס, ונותר חלק הרשת והתת-רשת. דרכים לייצוג mask:

1. יצוג בינרי –

11111111 11111111 11111100 00000000

2. יצוג דצימלי – 255.255.252.0

3. ציון אורך חלק הרשת (מספר ה-1-ים) - /22

- החלוקה למחלקות גורמת לבזבז כתובות IP.
- המחסור בכתובות IP הולך ומחריף.
- גוברת הדרישה לחיבור קבוע לכל אורך היום של גופים מסחריים שונים ואף לקוחות פרטיים (המחוברים לאינטרנט בכבלים ובאמצעות טכנולוגיית ADSL).

CIDR – Classless InterDomain Routing

- הפתרון – לבטל את החלוקה לקלאסים ולהקצות כתובות כנדרש, תוך יישום טכניקת מיסוך (masking).
לדוגמה, נניח כי כתובת הרשת 194.24 (ממחלקה C) פנויה.
- היא מכילה 8 סיביות בשדה כתובת המחשב (כלומר, עד 256 מחשבים בלבד).
- זה לא יתאים לארגון שזקוק ל- 2000 כתובות.
בעזרת שיטת CIDR הארגון מקבל בלוק של 2048 כתובות, למשל 194.24.0.0 עד 194.24.7.255, וכדי להבחין בין הרשת של הארגון לבין שאר הרשתות המתחילות ב- 194.24 מצורפות אליו מספר סיביות המיסוך, והדבר נכתב כ- 194.24.0.0/21.

מבוא לרשתות תקשורת מחשבים – פרק 5

CIDR – Classless InterDomain Routing

דוגמא: כל מרחב הכתובות המתחיל ב-194.24.0.0 פנוי.

אוניברסיטת קיימברידג' מבקשת 2048 כתובות,

אדינבורו – 1024, ואוקספורד – 4096.

| University | First address | Last address | How many | Written as |
|-------------|---------------|---------------|----------|----------------|
| Cambridge | 194.24.0.0 | 194.24.7.255 | 2048 | 194.24.0.0/21 |
| Edinburgh | 194.24.8.0 | 194.24.11.255 | 1024 | 194.24.8.0/22 |
| (Available) | 194.24.12.0 | 194.24.15.255 | 1024 | 194.24.12/22 |
| Oxford | 194.24.16.0 | 194.24.31.255 | 4096 | 194.24.16.0/20 |

Figure 5-59. A set of IP address assignments.

The routing tables all over the world are now updated with the three assigned entries. Each entry contains a base address and a subnet mask. These entries (in binary) are:

| Address | Mask |
|--|-------------------------------------|
| C: 11000010 00011000 00000000 00000000 | 11111111 11111111 11111000 00000000 |
| E: 11000010 00011000 00001000 00000000 | 11111111 11111111 11111100 00000000 |
| O: 11000010 00011000 00010000 00000000 | 11111111 11111111 11110000 00000000 |

תהליך הניתוב:

- עבור כל ארגון יש רשומה בטבלה שמכילה את כתובתו ואת ה-mask.
- מבצעים and לכתובת היעד של החבילה עם ה-mask, ומשווים לכתובת הבסיס.

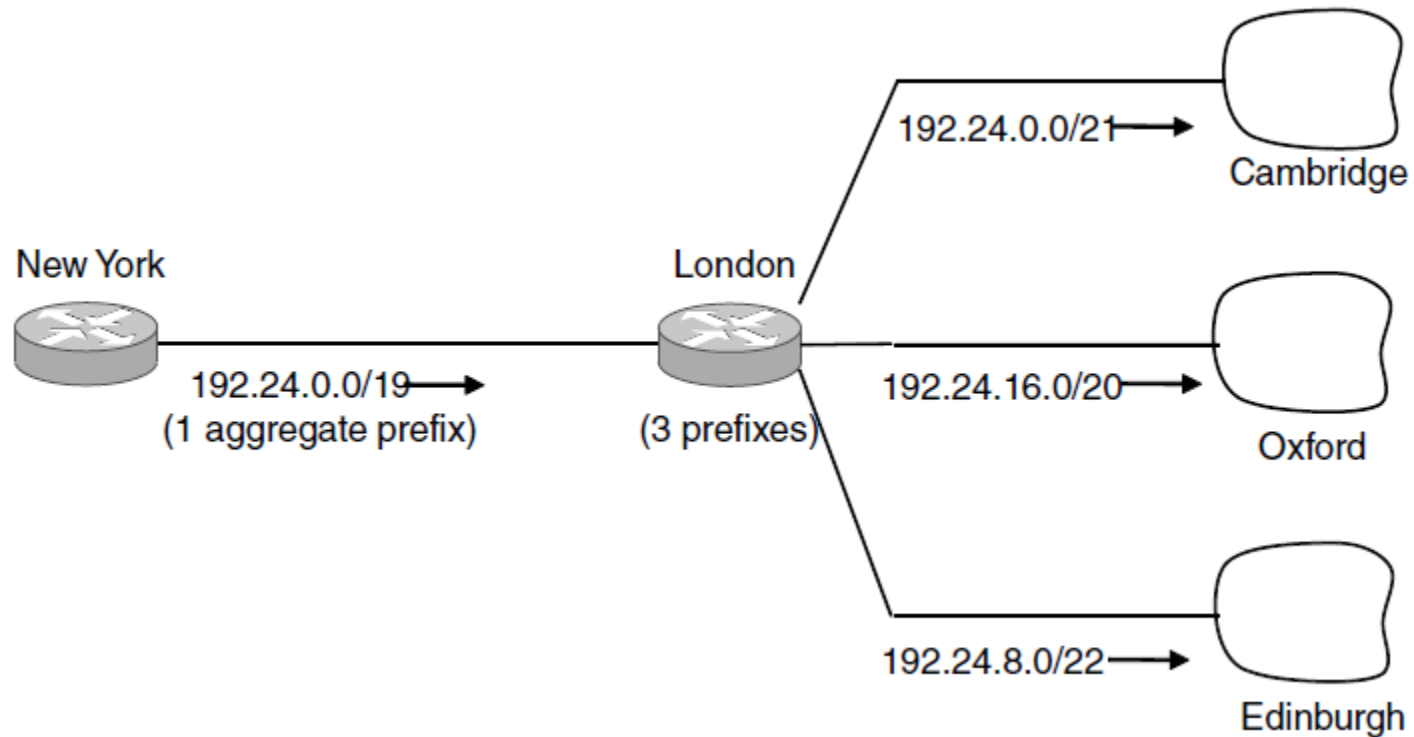
דוגמא:

- מגיעה חבילה עם הכתובת: 194.24.17.4 ששקולה ל-
11000010.00011000.00010001.000000100
- מבצעים and עם ה-mask של קיימברידג' – לא מתאים.
- מבצעים and עם ה-mask של אדינבורו – לא מתאים.
- מבצעים and עם ה-mask של אוקספורד – מתאים.

Aggregation

- אם במקום מסוים יש קו יוצא יחיד עבור אנגליה, ניתן לאחד את שלוש השורות לאחת: 194.24.0.0/19
- הכתובת הראשונה בטווח הכתובות הזה:
194.24.000000000.000000000
- והאחרונה: 194.24.00011111.11111111
- החלק הקבוע הוא 19 סיביות.

aggregation



Aggregation

אם בהמשך מקצים את החלק הפנוי לסן-פרנסיסקו,

מוסיפים שורה 22 / 194.24.12.0

הניתוב מתבצע ע"י הקידומת המתאימה הארוכה ביותר –

.Longest matching prefix routing

Network Address Translation – NAT

השיטה:

- לארגון עם משתמשים רבים תוקצה כתובת IP אחת.
- בתוך הארגון, כל מחשב יקבל כתובת IP ייחודית.
- כשמחשב בארגון שולח חבילה, מוחלפת כתובת ה-IP של המקור בכתובת ה-IP של הארגון.
- מספר ה-PORT של המקור מוחלף באינדקס כלשהו בטבלת התרגום.
- השורה המיוצגת על ידי האינדקס הנ"ל מכילה את כתובת ה-IP של המקור ואת מספר ה-PORT של המקור.

NAT

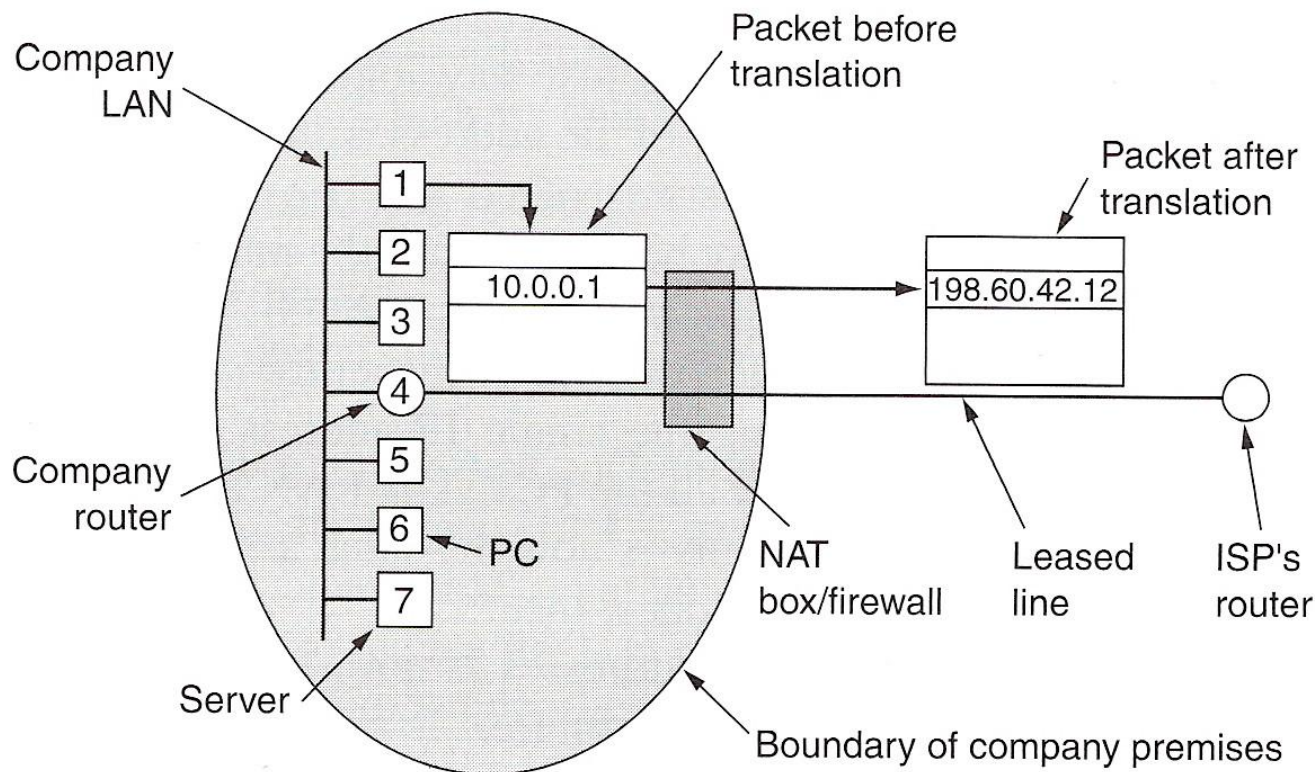
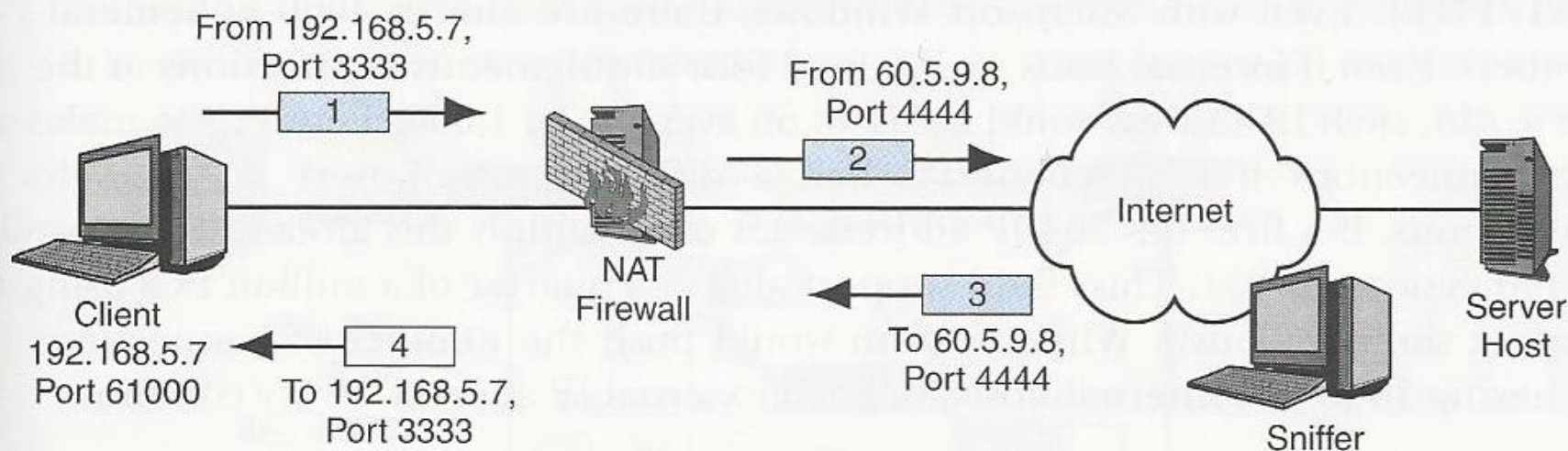


Figure 5-60. Placement and operation of a NAT box.

NAT



Translation Table

| Internal | | External | |
|-------------|------|----------|------|
| IP Addr | Port | IP Addr | Port |
| 192.168.5.7 | 3333 | 60.5.9.8 | 4444 |
| ... | ... | ... | ... |

Figure 10-13 Network Address Translation (NAT)

Network Address Translation – NAT

- כאשר מתקבלת חבילה בחזרה למחשב השולח, מבצע NAT תרגום הפוך:
1. ניגש לשורה המתאימה בטבלה (שורה זו תופיע בחבילה כ-PORT יעד).
 2. השורה המיוצגת על ידי האינדקס הנ"ל מכילה את כתובת ה-IP של היעד (מי שהיה המקור), ואת מספר ה-PORT שלו.

Network Address Translation –NAT

The only rule is that no packets containing these addresses may appear on the Internet itself. The three reserved ranges are:

| | | |
|-------------|----------------------|--------------------|
| 10.0.0.0 | – 10.255.255.255/8 | (16,777,216 hosts) |
| 172.16.0.0 | – 172.31.255.255/12 | (1,048,576 hosts) |
| 192.168.0.0 | – 192.168.255.255/16 | (65,536 hosts) |

חסרונות:

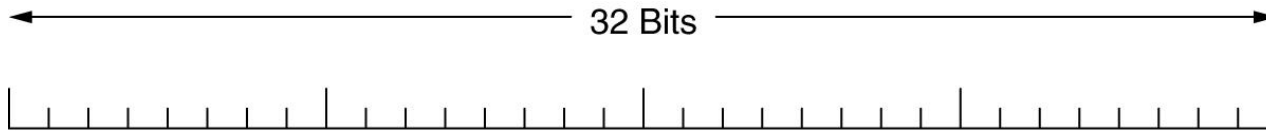
- הפרת העיקרון שכל כתובת IP מציינת מחשב אחד בלבד.
- במקרה של קריסת ה-NAT Box, לא ניתן יהיה לשחזר לאיזה מחשב שייכות החבילות הנכנסות.
- השיטה מפרה את מודל השכבות של האינטרנט.
- פגיעה ביישומים שאינם משתמשים ב-TCP/UDP אלא בפרוטוקול תובלה אחר.
- פגיעה בפרוטוקולים המסתמכים על כתובת ה-IP של המקור במקומות אחרים ולא רק ב-Header (כגון FTP).

IPv6 – החידושים

- כתובות ארוכות יותר – 128 סיביות במקום 32.
- מבנה כותרת גמיש – הכותרת הבסיסית היא בעלת אורך קבוע, אך קיימת תמיכה בהרחבות.
- **QoS** – ניתן לדרג מנות לפי סדר קדימות.
- **Anycast** – סוג מיעון חדש המתייחס לכתובת המשותפת לכמה מחשבים, כאשר יש לשדר את המסר לאחד בלבד.
- אבטחה משופרה.
- תמיכה במסגרות ארוכות מ- 64KB – **Jumbograms**.

מבוא לרשתות תקשורת מחשבים – פרק 5

IPV6



| | | | |
|-----------------------------------|---------------|-------------|-----------|
| Version | Traffic class | Flow label | |
| Payload length | | Next header | Hop limit |
| Source address (16 bytes) | | | |
| Destination address (16 bytes) | | | |

IPV6 – Extension Headers

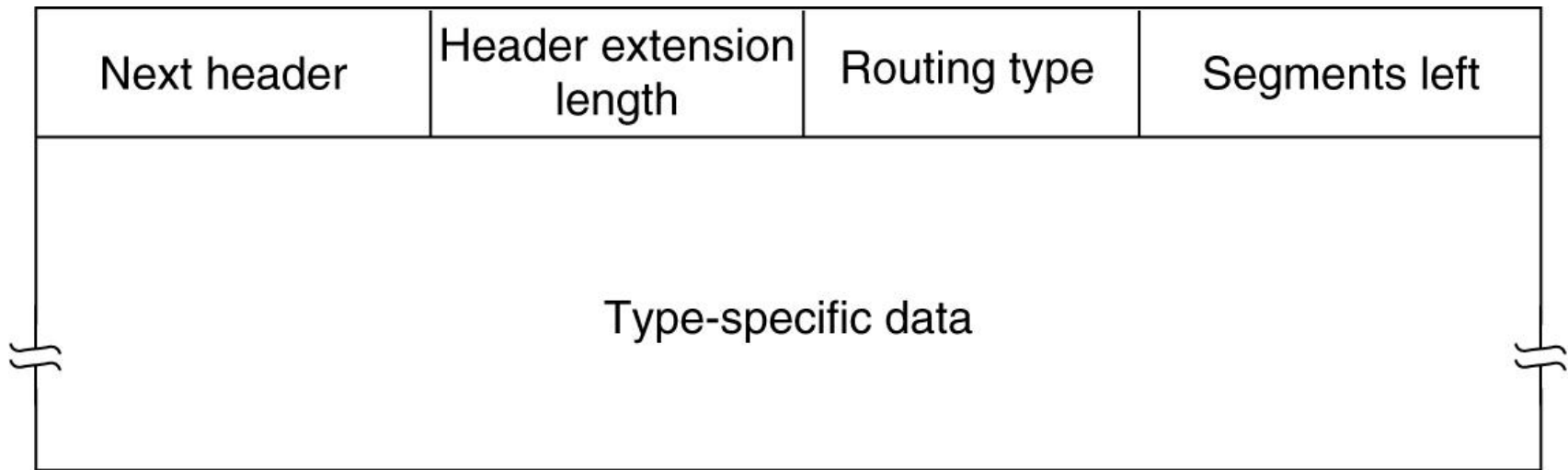
| Extension header | Description |
|----------------------------|--|
| Hop-by-hop options | Miscellaneous information for routers |
| Destination options | Additional information for the destination |
| Routing | Loose list of routers to visit |
| Fragmentation | Management of datagram fragments |
| Authentication | Verification of the sender's identity |
| Encrypted security payload | Information about the encrypted contents |

IPV6 – Extension Headers (2)

| | | | |
|----------------------|---|-----|---|
| Next header | 0 | 194 | 4 |
| Jumbo payload length | | | |

The hop-by-hop extension header for large datagrams (jumbograms)

IPV6 – Extension Headers (3)

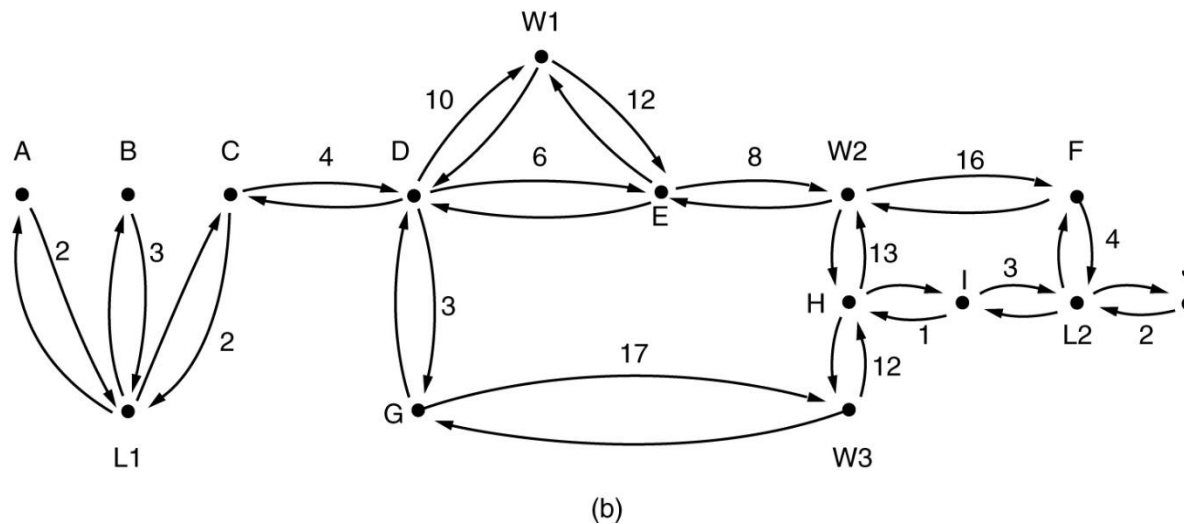
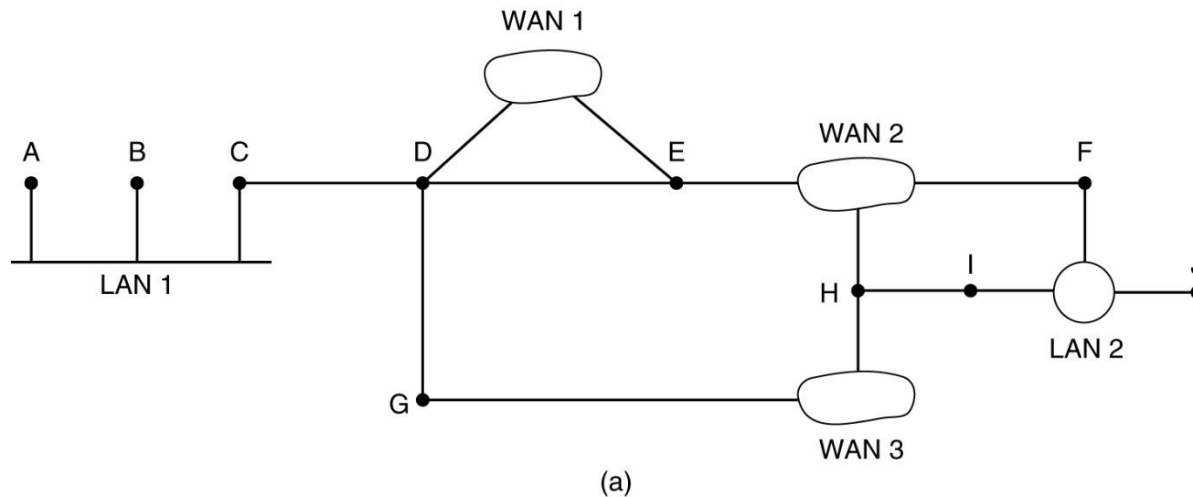


The Extension Header for routing

פרוטוקול Open Shortest Path First – OSPF

- פרוטוקול ניתוב פנים-רשתי.
- הרשת המקומית מחולקת לאזורים המקושרים באמצעות רשת שלדית (אזור 0).
- הרשת השלדית מורכבת מנתבים פנים שלדיים Backbone Routers, המקשרים בין צומתי הרשת השלדית, ומנתבי גבול אזור Area Border Routers, המקשרים בין הרשת השלדית לאזורים אחרים בתחום המערכת העצמאית.
- האזורים האחרים מורכבים מ:
 - נתבים פנים אזוריים Internal Routers המקשרים בין צומתי האזור
 - נתבי חוץ AS Boundary Routers המקשרים בין מערכת עצמאית אחת לאחרת.

פרוטוקול Open Shortest Path First – OSPF

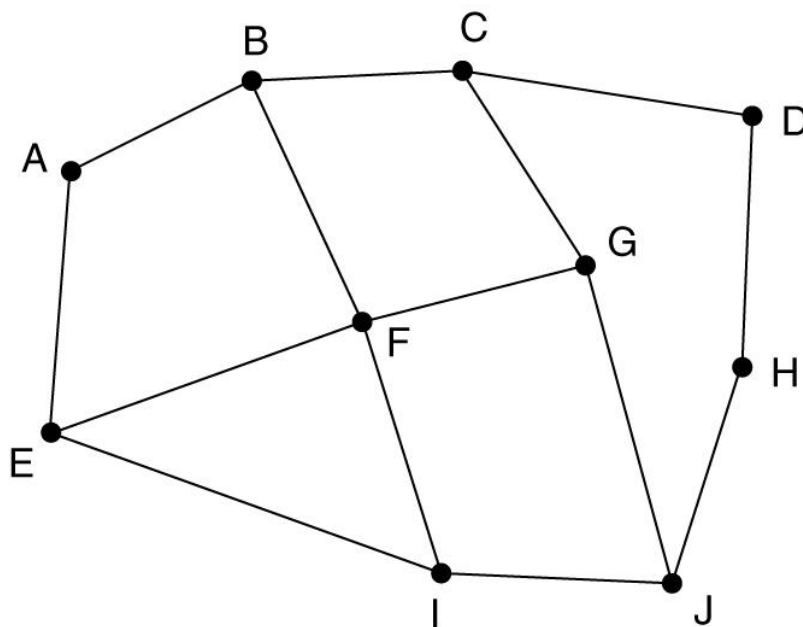


(a) An autonomous system. (b) A graph representation of (a).

פרוטוקול Border Gateway Protocol –BGP

- פרוטוקול ניתוב בין-רשתי.
- גירסה מורחבת של פרוטוקול וקטורי מרחק.
- בכל שורה שומרים גם את המסלול המלא ליעד.
- לא קיימת בעית הספירה לאינסוף.

פרוטוקול Border Gateway Protocol –BGP



(a)

Information F receives
from its neighbors about D

From B: "I use BCD"
From G: "I use GCD"
From I: "I use IFGCD"
From E: "I use EFGCD"

(b)

(a) A set of BGP routers. (b) Information sent to F.