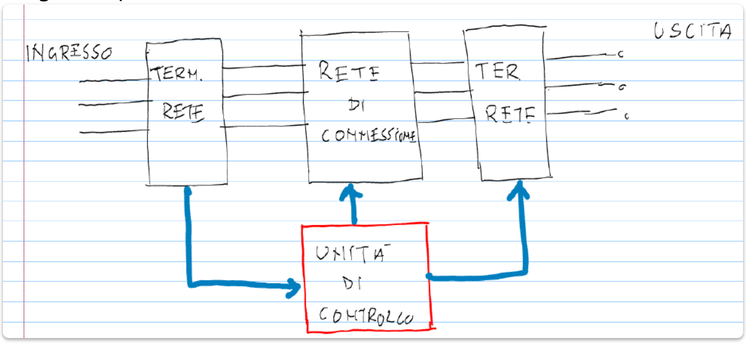
STRUTTURA DI UN AUTOCOMMUTATORE

Un autocommutatore è preposto alla commutazione, cioè all'inoltro d'informazioni da una sorgente verso un certo destinatario.
È composto dai seguenti elementi, collegati in questo modo:



L'autocommutatore è un dispositivo intelligente. Permette d'interpretare (autonomamente) le richieste di connessione provenienti dalla sorgente grazie all'unità di controllo (parte "decisionale" SW). Essa poi interagisce con la rete di connessione (parte fisica HW che consente il collegamento effettivo tra un punto d'ingresso e un punto di uscita)
Studieremo in particolare le reti di connessione, come sono fatti internamente

> *Una volta in assenza di unità di controllo il collegamento veniva effettuato a mano da degli addetti, che ascoltavano a voce la richiesta dell'utente in ingresso e chiudevano il collegamento con il desiderato utente di uscita. In tali casi si parlava di commutatore, perché l'inoltro era manuale (non autonomo/automatico)*

Come si nota in figura, sono presenti anche i *blocchi di terminazione*, in ingresso e in uscita: in essi arrivano delle linee dall'esterno per poi ripartire verso altre destinazioni (blocchi di front-end).

- Nel caso lato sorgente, servono per individuare/dividere dal canale di ingresso le *informazioni di segnalazione*: comandi che saranno interpretati dalla unità di controllo per attuare determinate azioni sul blocco centrale, ovvero sulla rete di connessione. Si esegue cioè una operazione di **demultiplexing**
- Lato destinazione, viene reimmessa nel segnale l'informazione di segnalazione secondo le regole presenti nella unità di controllo per poter essere poi inoltrata. Viene eseguita cioè una multiplazione (**multiplexing**)

TIPI DI COMMUTATORI

Indicando con n_{IN} le linee di ingresso e con n_{OUT} quelle di uscita, si hanno i seguenti casi:

- $n_{IN} = n_{OUT} \rightarrow \text{Distributori}$
- $n_{IN} > n_{OUT} \rightarrow \text{Concentratori}$
- $n_{IN} < n_{OUT} \rightarrow \text{Espansori}$

RETI DI CONNESSIONE

Esistono due alternative:

- A divisione di **spazio** (S)
- A divisione di **tempo** (T) oppure (TSI)

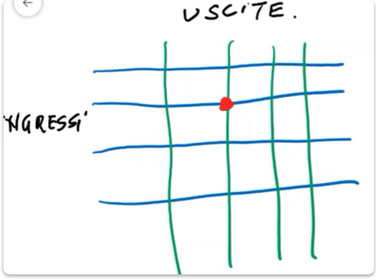
ALTRE CARATTERISTICHE DI UNA RETE DI CONNESSIONE

Oltre alla divisione S e T, in generale una rete di connessione si caratterizza anche a seconda di:

- Multistadio (mono): esistono infatti reti che congiungono più stadi S e T insieme (esempio: TST), fino a 5 stadi totali
- Fattore di costo (legato al numero totale di chiusure di collegamenti che la struttura deve fare)
- Bloccante (non): quando la struttura non riesce a gestire tutte le richieste (per vari motivi). Ci occuperemo però di strutture *non* bloccanti (cfr. struttura T-S)

DIVISIONE DI SPAZIO (S) --> cambio di linea

Le prime implementate storicamente. Semplici da realizzare/spiegare:



Si vuole stabilire una connessione (elettrica) tra linea di ingresso e linea di uscita desiderata. I punti di incrocio della matrice sono i punti di collegamento, che saranno poi successivamente deallocati quando l'utilizzo finisce

- Linee blu: ingresso
- Linee verdi: uscite

> *In questo caso quindi la commutazione consiste nello stabilire continuità elettrica tra ingressi e uscite*

😃: sono utilizzabili sia nel caso di telefonia analogica che numerica (versatilità);
😡: nel caso di telefonia numerica il collegamento rimane attivo solo per il tempo di trasmissione degli $8\mathrm{bit}$, quindi da un punto di vista di stress è più impegnativo, dato che ogni $125\ \mu s$ la matrice si deve riconfigurare

FATTORE DI COSTO

Se N sono le linee d'ingresso e M sono le linee di uscita, allora il fattore di costo (C) è:

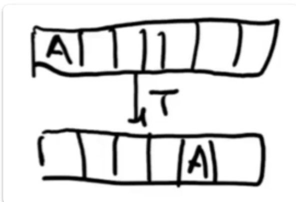
$$\boxed{C = N \cdot M}$$

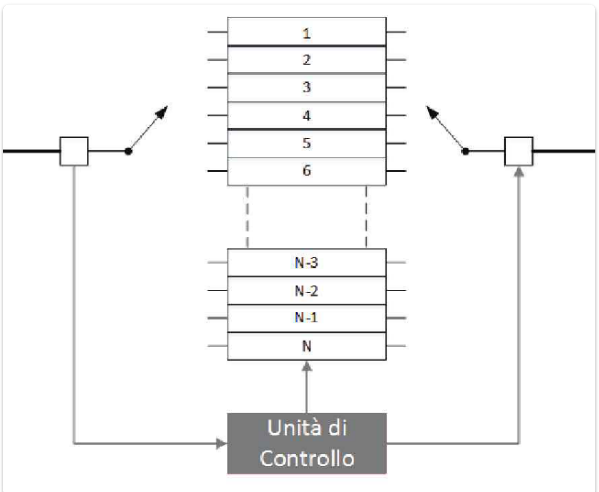
> *Nel caso in cui $N = M$, allora: $C = N^2$*

DIVISIONE DI TEMPO (T) --> permutazione di canale

Sono delle memorie: serie di celle ciascuna delle quali permette la memorizzazione di $8\ \mathrm{bit}$. L'accesso in lettura e scrittura a tale memoria si rappresenta con un commutatore, che riceve le informazioni/comandi da una unità di controllo centrale (logica di controllo)

- Si esegue un cambio di posizione (permutazione) della posizione dei canali nella stessa trama di appartenenza. In figura è riportato il cambio del canale A in una trama formata da 7 canali





😡: solo nella telefonia numerica

MODALITA' DI PERMUTAZIONE

- Scrittura (accesso) sequenziale e lettura casuale: i canali vengono inseriti in memoria mantenendo il loro ordinamento iniziale (i-esimo gruppo di $8\ \mathrm{bit}$ viene scritto nella i-esima cella della memoria). La lettura (ovvero il trasferimento dei byte dalla memoria alla trama di uscita) non mantiene in generale l'ordinamento iniziale ma permuta secondo alcune l'ordine che vogliamo in uscita (di dice "casuale" perché non è deterministica). *La commutazione si realizza in lettura*
- Scrittura casuale e lettura sequenziale: in ingresso si inseriscono in memoria i canali nella posizione che vogliamo in uscita (eseguendo eventuali permutazioni) e poi in lettura si scansionano in sequenza le celle della memoria per giungere poi alla trama di uscita. *La*

commutazione si realizza in scrittura

FATTORE DI COSTO

Relativo alla velocità con cui si può accedere alla memoria in lettura e in scrittura (t_a)

- In particolare, durante un singolo tempo di slot/trama ($125\ \mu s$) si devono eseguire le operazioni di lettura (degli 8 bit) e scrittura (degli 8 bit) in maniera completa per ogni canale.
- Supponiamo di avere N canali in ingresso e N canali in uscita (quindi $2N$ canali in tutto da gestire)
  Abbiamo come tempo di accesso t_a:
  $$t_a = \frac{125\ \mu s}{2N}$$
- Supponendo di avere un numero diverso di canali in ingresso e in uscita, ovvero $N \neq M$, $N > M$, allora abbiamo (caso peggiore)
  $$\boxed{C = t_a \leq \frac{125\ \mu s}{2N}}$$
  (maggiore è il numero di canali, maggiore è il tempo di accesso. Dopo un certo limite, il tempo diventa troppo alto e quindi sono state pensate delle alternative più efficienti)

> **Sia le strutture S che le strutture T sono non bloccanti**

STRUTTURE MULTISTADIO BISTADIO

Vorremmo perseguire i seguenti 3 obiettivi nella realizzazione di una rete di connessione (progetto):

- Aumentare funzionalità della struttura
- Ridurre i costi
- Avere un comportamento non bloccante

Le strutture monostadio, permettono in un certo momento una sola delle operazioni: o cambio di linea o cambio di canale. Vogliamo realizzare una struttura che include entrambe le funzioni

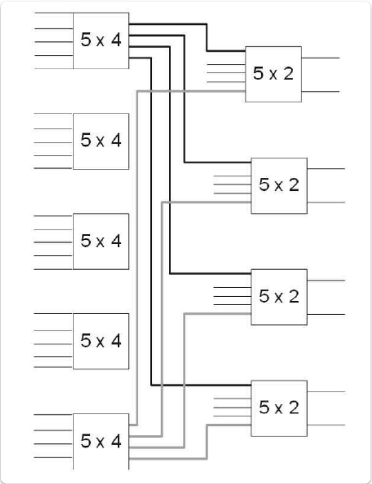
TIPOLOGIE

- Omogenee: tutte S o tutte T
- Non omogenee mix di S e T

MULTISTADIO S-S (omogenea)

Due stadi S (primo e secondo)

- Il primo stadio è composto da un certo numero di gruppi. Su ciascuno di essi giunge lo stesso numero di linee. Nell'esempio di progetto in figura abbiamo 25 ingressi e 5 gruppi: pertanto su ciascun gruppo convergono 5 linee
- Per il secondo stadio si ripete lo stesso ragionamento: è composto da un certo numero di gruppi su cui in ognuno sono collegati lo stesso numero di linee. Nell'esempio abbiamo 4 uscite provenienti da ciascun gruppo del primo stadio che diventano gli ingressi per i gruppi del secondo stadio. In particolare saranno proprio 4 i gruppi, da ciascuno dei quali usciranno 2 linee, quindi in tutto abbiamo 8 linee di uscita ($\star$).
  - Nel dettaglio, *l'uscita j del gruppo i del primo stadio giunge all'ingresso i dell gruppo j del secondo stadio*: ad esempio, la terza uscita del primo gruppo del primo stadio viene connessa al terzo ingresso del primo gruppo del secondo stadio
    In altre parole: $1^\circ\ \underbrace{(x,y)}_{\text{(blocco,linea)}} \longrightarrow 2^\circ\ \underbrace{(y,x)}_{\text{(blocco,linea)}}$



($\star$) abbiamo avuto questi numeri nell'ipotesi di voler fare un progetto che permette di passare da 25 ingressi a 8 uscite

😃: costi complessivi minori (vedi dopo)
😡: S-S è una struttura *bloccante*

FATTORE DI COSTO

Guardiamo il caso del progetto $25 \to 8$

> *Come sappiamo, se avessimo una struttura* S *monostadio essa avrebbe costo:* $C = 25 \cdot 8 = 200$

Per la struttura multistadio, eseguiamo la *somma dei singoli stadi*, quindi:

$$\begin{array}{lccc} 1^\circ \text{ stadio:} & 5 \cdot 20 & = & 100 \\ 2^\circ \text{ stadio:} & 4 \cdot 10 & = & 40 \\ \text{TOTALE} & 100 + 40 & = & 140 \quad \checkmark \end{array}$$

Abbiamo una riduzione di prezzo rispetto alla monostadio e quindi abbiamo soddisfatto il primo requisito fondamentale

LA STRUTTURA È BLOCCANTE? (3° REQUISITO) - SPOILER: SI :(

La struttura è bloccante perché non possiamo rispettare tutte le possibili richieste che possono avvenire: *esiste una sola connessione che connette un blocco di uno stadio a un blocco dello stadio successivo*, quindi una nuova richiesta in ingresso allo stesso blocco di partenza che necessita di connettersi con il blocco successivo in una linea diversa da quella a cui è attualmente connessa, causa un blocco. In altri termini, facendo un esempio specifico:

UNA SECONDA RICHIESTA IN INGRESSO AL PRIMO BLOCCO S PER I STADIO CHE NECESSITA DI ESSERE CONNESSA CON LA SECONDA LINEA LIBERA DEL PRIMO BLOCCO DEL II STADIO VIENE BLOCCATA.

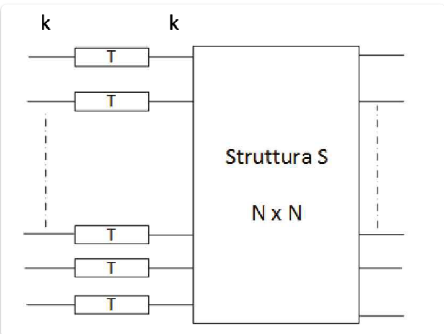
Per evitare il blocco, sarebbe necessario aumentare il numero di connessioni tra elementi del primo stadio verso ogni elemento del secondo stadio fino al punto in cui esso risulti pari o superiore al numero di uscite dei blocchi del secondo stadio
Dimostriamo però che tale soluzione peggiora i costi della struttura, infatti:

$$\begin{array}{lccc} 1^\circ \text{ stadio:} & 5 \cdot (5 \cdot 8) & = & 200 \\ 2^\circ \text{ stadio:} & 4 \cdot (10 \cdot 2) & = & 80 \\ \text{TOTALE} & 200 + 80 & = & 280 \quad \times \end{array}$$

MULTISTADIO T-S (non omogenea)

La struttura multistadio non omogenea per sua definizione porta vantaggi in termini di numero di funzionalità rispetto a quelle monostadio. In particolare la non omogeneità permette allo stesso tempo di:

- Cambiare canale $\to$ grazie allo stadio T
- Cambiare linea $\to$ grazie allo stadio S



Facciamo un esempio di funzionamento per capire meglio le funzionalità.

$$\text{Canale } 8 \text{ linea } 1 \xrightarrow{\text{vuole andare}} \text{Canale } 12 \text{ linea } 7$$

- Il primo blocco che si incontra è il T, cioè a divisione di tempo: esso esegue il cambio di canale in quello corretto, mantenendo intatta la linea, quindi la configurazione parziale risulta:

$$\text{Canale } \cancel{8}^{12} \text{ linea } 1$$

- Dopodiché con l'incontro del blocco S si esegue il cambio di linea e si conclude, infatti risulta:

$$\text{Canale } 12 \text{ linea } \cancel{1}^{7}$$

ovvero

$$\text{Canale 12 linea 7}$$

😃: riesco a eseguire due operazioni contemporaneamente che singolarmente con le strutture monostadio risulterebbero impossibili
😡: la struttura è **bloccante** ($\triangle$)

$\triangle$: per dimostrare che è bloccante è sufficiente trovare una richiesta che non può essere realizzate mediante questa struttura.
Si ha conflitto ad esempio in questa situazione, vogliamo eseguire le seguenti richieste:

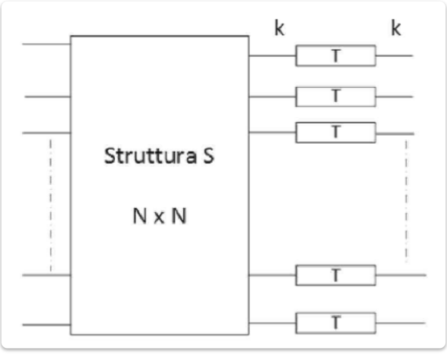
$$\begin{gathered}\text{Canale 7 linea 1} \longrightarrow \text{ Canale 8 linea 3} \\ \text{Canale 12 linea 1} \longrightarrow \text{Canale 8 linea 2}\end{gathered}$$

Abbiamo due richieste che arrivano sulla stella linea di ingresso (canali diversi) che vogliono andare ad occupare lo stesso canale di destinazione (a linee differenti)
La struttura T permette il cambio di canale verso il canale 8, ma poi la struttura S non riesce a gestire la richiesta completa circa il cambio di linea, perché può accogliere una sola delle due

> *in generale la struttura è bloccate ogni qualvolta che sulla stessa linea di ingresso su canali diversi si presentano richieste di connessione su linee diverse sullo lo stesso canale*

STRUTTURA S-T

In maniera simmetrica rispetto al caso precedente:



Stesse caratteristiche formali della precedente, vediamo con un esempio come si comporta:

$$\text{Canale 8 linea 1} \longrightarrow \text{ Canale 12 linea 7}$$

- La struttura S permette il cambio di linea e la T il cambio di canale, quindi:

$$\text{Canale 8 linea 1} \xrightarrow{\mathrm{S}} \text{ Canale 8 linea 7} \xrightarrow{\mathrm{T}} \text{ Canale 12 linea 7}$$

  😡: Struttura **bloccante** ($\triangledown$)

$\triangledown$: in maniera duale rispetto al caso precedente, analizziamo il caso:

$$\begin{gathered}\text{Canale 7 linea 1} \longrightarrow \text{ Canale 12 linea 4} \\ \text{Canale 7 linea 3} \longrightarrow \text{Canale 8 linea 4}\end{gathered}$$

Anche in questo caso si può eseguire una richiesta sola contemporaneamente

> *in generale la struttura è bloccate ogni qualvolta che sullo stesso canale di ingresso e linee diverse si presentano richieste di connessione su canali diversi e stesse linee*

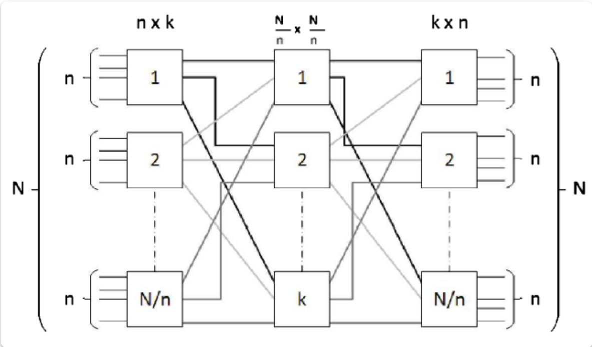
STRUTTURE MULTISTADIO TRISTADIO

STRUTTURA S-S-S (omogenea)

Idea: dato che non aumentano i gradi di libertà essendo la struttura omogenea, ci limitiamo a eseguire il cambio di linea
Supponiamo:

- N linee ingresso
- N linee uscita
- $\dfrac{N}{n}$ gruppi (blocchi) in cui si dividono le linee di ingresso/uscita
  - Ogni blocco ha quindi n linee
- Primo stadio: $\frac{N}{n}$ blocchi con n linee d'ingresso e k linee di uscita (blocchi $n \times k$);

- Secondo stadio: k blocchi con $\frac{N}{n}$ linee d'ingresso e $\frac{N}{n}$ linee di uscita (blocchi $\frac{N}{n} \times \frac{N}{n}$);
- Terzo stadio: $\frac{N}{n}$ blocchi con k linee d'ingresso e n linee di uscita (blocchi $k \times n$, simmetrico al primo)



Si noti che il numero di linee di uscita del blocco i, corrisponde al numero di linee d'ingresso del blocco $i+1$.
Nota: $k >> n$ per ipotesi (cfr. Formula di Clos)

FATTORE DI COSTO

Il costo complessivo della S-S-S è dato dalla somma dei costi dei singoli stadi, quindi:

$$\begin{aligned} C_{TOT} &= C_I + C_{II} + C_{III} \\ &= \frac{N}{n} \cdot (n \cdot k) + k \cdot \left(\frac{N}{n} \cdot \frac{N}{n}\right) + \frac{N}{n} \cdot (k \cdot n) \\ &= 2 \cdot \left(\frac{N}{n} \cdot (n \cdot k)\right) + k \cdot \left(\frac{N^2}{n^2}\right) \\ &= 2 \cdot (N \cdot k) + \frac{k \cdot N^2}{n^2} \\ &\boxed{= (N \cdot k) \cdot \left(2 + \frac{N}{n^2}\right)} \end{aligned}$$

Dobbiamo quindi cercare dei valori per n e k tali che:

- il costo sia inferiore alla monostadio S
- *la struttura sia non bloccante*
  Questo è permesso dalla **formula di Clos**

FORMULA DI CLOS

Approccio: dimostrare che la struttura è non bloccante con determinati valori nel caso peggiore.

- Metodo di analisi del caso peggiore

Andiamo dunque a vedere quando si verifica il caso peggiore, basandosi su alcune ipotesi preliminari. Quindi supponiamo che:
Ipotesi:

1.

$n-1$ linee d'ingresso occupate del blocco i del primo stadio;
$n-1$ linee d'uscita occupate del blocco j del terzo stadio;
Si vuole effettuare quindi il collegamento tra l'unica linea ingresso disponibile del blocco i del primo stadio con l'unica linea uscita disponibile del blocco j del terzo stadio

2.

Il primo e il terzo stadio sono insiemi disgiunti, cioè non hanno elementi in comune: nessun ingresso al blocco i del primo stadio sarà una uscita del medesimo blocco i del terzo stadio.

> *Se si parte da un blocco i si andrà sicuramente al terzo stadio a un blocco $j \neq i$*

3.

Le $n-1$ linee occupate in ingresso al blocco di riferimento del primo stadio sono indirizzate verso blocchi del secondo stadio diversi da quelli da cui partono i collegamenti che generano le uscite occupate nel terzo stadio

MINIMIZZARE k

Si dimostra che, il valore di k minimo per rispettare le richieste è:

$$\boxed{k = 2n - 1} \qquad \text{(formula di Clos)}$$

Dimostrazione:
Dobbiamo trovare un percorso adatto con le ipotesi fatte. Iniziamo con il dire che:

- L'ingresso di riferimento (quello libero che vogliamo mandare in uscita) non potrà essere inviato agli $n-1$ blocchi che hanno già ingressi occupati dalle altre $n-1$ richieste presenti in ingresso. Quindi (rimane un solo blocco):

$$\underline{k \geq n - 1}$$

- Dato che le uscite devono essere per ipotesi disgiunte dagli ingressi, allora non si potrà indirizzare la richiesta verso qualsiasi blocco del II stadio. In particolare dobbiamo escludere quelli che hanno le uscite tali da non far rispettare l'ipotesi di disgiunzione. Quindi vanno tolti altri $n-1$, pertanto:

$$k \geq \overbrace{(n-1)}^{1^\circ \text{ ipotesi}} + \overbrace{(n-1)}^{2^\circ \text{ ipotesi}} \Rightarrow k \geq 2(n-1) \Rightarrow \underline{k \geq 2n - 2}$$

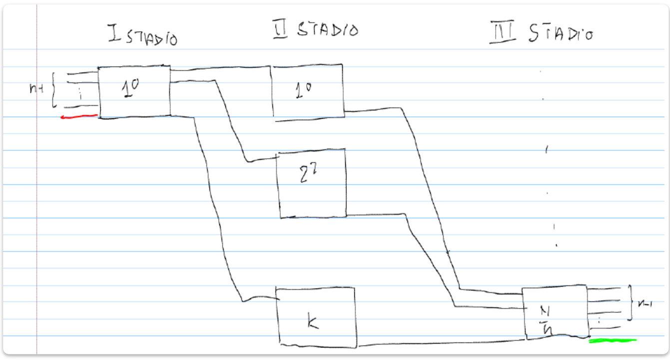
- Per chiudere, dobbiamo disporre di almeno un altro blocco che deve avere ingresso libero verso un blocco del primo stadio (così da poter fare il collegamento) e uscita libera verso un blocco del terzo stadio. Quindi si somma $+1$, ottenendo la formula

$$k \geq 2n - 2 + 1 \Rightarrow \underline{k \geq 2n - 1}$$

  Prendendo il valore minimo quindi:

$$\boxed{k = 2n - 1} \quad , \quad \text{C.V.D}$$

  Un possibile percorso è il seguente:



Andando a sostituire tale valore con la formula del costo generale ottenuta precedentemente, si ha:

$$\begin{aligned} C_{TOT} = \quad (N \cdot k) \cdot \left(2 + \frac{N}{n^2}\right) \quad &= \qquad 2Nk + k\frac{N^2}{n^2} \\ &= \quad 2N(2n-1) + (2n-1)\frac{N^2}{n^2} \end{aligned}$$

- Questo è il costo minimo per rendere la struttura S-S-S non bloccante

MINIMIZZARE n

Ci chiediamo ora, essendo n un parametro da scegliere: *qual è il valore ottimo di n che minimizza la funzione di costo?* In questo modo completiamo l'analisi.
Dato che queste strutture sono chiamate a gestire un numero alto di linee, quindi N è elevato.

- Di conseguenza seppur più piccolo, sarà grande anche n
  Vale perciò la relazione:

$$N >> 1 \implies \boxed{n >> 1}$$

  Ponendo l'ultima relazione nella formula C_{TOT}, si conclude che, asintoticamente:

$$C \approx 4Nn + \frac{2N^2}{n}$$

  Calcolando la derivata del costo rispetto a n (quindi trattando N come una costante) e ponendola a zero si ottiene (applicando la linearità):

$$\frac{dC}{dn} = 4N - \frac{2N^2}{n^2} = 0$$

  da cui:

$$n = \sqrt{\frac{N}{2}}$$

Basta adesso sostituire quest'ultimo valore di n e quello di k nella equazione del costo per concludere il discorso:

$$\boxed{C_{\text{ottimo}} = 4\sqrt{2}N^{3/2}}$$

che è un valore minore rispetto a C_S, ovvero al costo della struttura monostadio ✓

Nota bene (VINCOLI)

Abbiamo calcolato la derivata come se avessimo una funzione continua, ma in realtà dobbiamo tenere presente che:

> *n e $\frac{N}{n}$ devono essere valori interi, in quanto rappresentano rispettivamente il numero di linee e il numero di blocchi*

Pertanto di conseguenza: $n = \sqrt{\frac{N}{2}}$ deve essere intero
Questo potrebbe causare dei limiti in fase di progetto, perché magari facendo i conti si scopre che i valori ottimi sarebbero non interi.
Allora si utilizza un approccio *euristico*, pertanto:

Soluzione (metodo euristico)

Si considerano il più grande valore inferiore al valore ottimo e il più piccolo valore superiore al valore ottimo che rispettano i vincoli architetturali.
Tra le due alternative si sceglie quel valore di n che genera il costo minimo

Facciamo un esempio per capire meglio:
Dato: $N = 10^5 \text{ linee}$
Utilizzando la formula di Clos, abbiamo

$$n_{\text{ott}} = \sqrt{\frac{N}{2}} = 222 \;\; \checkmark n \text{ è intero}$$

Tuttavia, si scopre presto che:

$$\frac{N}{n_{\text{ott}}} = 4545.45455 \;\; \text{X} \;\; \text{non è intero}$$

Si sceglie quindi: $n_1 = 200$, che porta

$$\frac{N}{n_1} = 500 \quad , \quad k = 399 \implies C = 1.8 \cdot 10^8$$

e $n_2 = 250$, che porta

$$\frac{N}{n_2} = 400 \quad , \quad k = 499 \implies C = 1.8 \cdot 10^8$$

Abbiamo ottenuto *una situazione particolare: entrambi i costi C sono uguali*
Dal punto di vista pratico però dipende dalla situazione logistica (fisica) circa i collegamenti che abbiamo da fare

Se avessimo utilizzato la struttura monostadio, avremmo ottenuto:

$$C \stackrel{\triangle}{=} N^2 = 10^{10}$$

ben due ordini di grandezza superiore
La formula di Clos è quindi ben più conveniente.