Università Degli Studi Roma Tre

Facoltà Di Scienze M.F.N. – Corso Di Laurea In Matematica

Tesi di Laurea in Matematica di

Irene Olivieri

Problemi di ottimizzazione combinatoria ed algoritmi per il physical mapping del DNA

Relatore
Prof. Marco Liverani

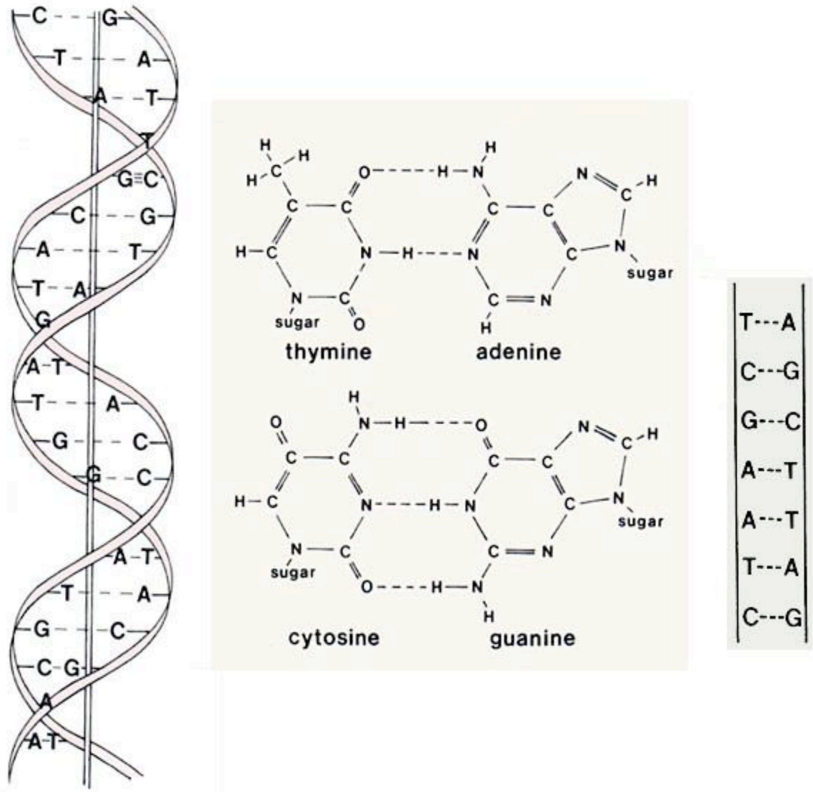
A.A. 2004 - 2005

Maggio 2006

DNA (acido deossiribonucleico)

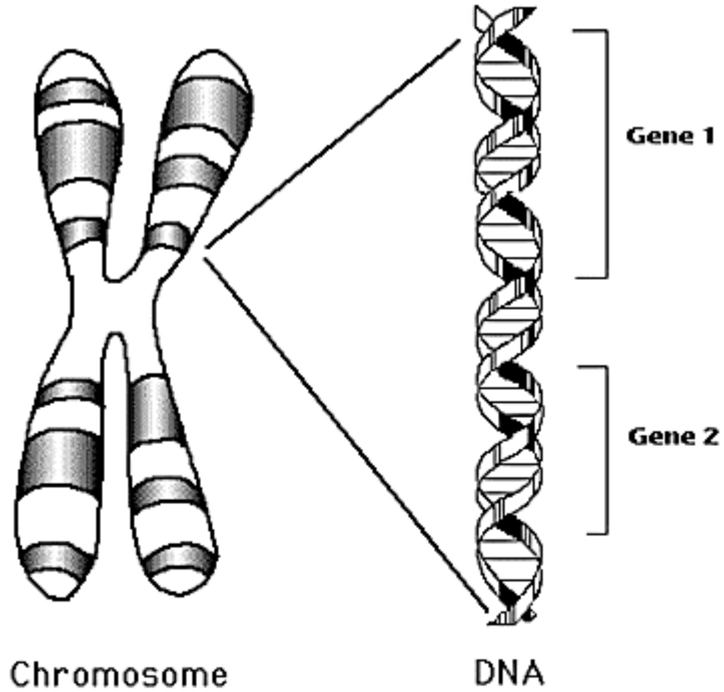
Doppia catena di **nucleotidi** composti da:

- deossiribosio (zucchero semplice)
- gruppo fosfato
- basi azotate:
  Adenina (**A**) ↔ Timina (**T**)
  Citosina (**C**) ↔ Guanina (**G**)



DNA nella cellula

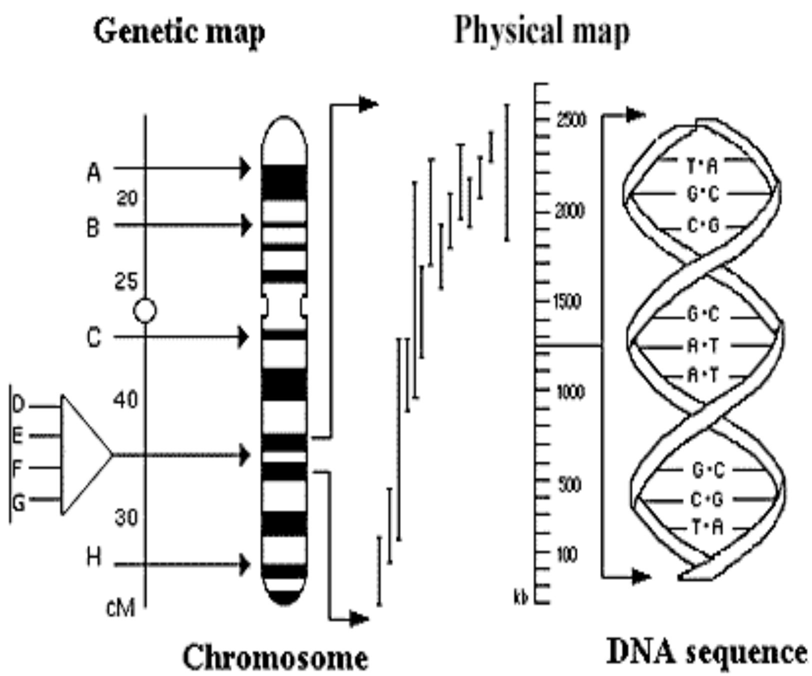
- **Cromosomi**
- **Geni**: tratti contigui di DNA che contengono le informazioni necessarie per la costruzione delle proteine
- **Genoma**: insieme completo dei cromosomi di una cellula



Physical Map

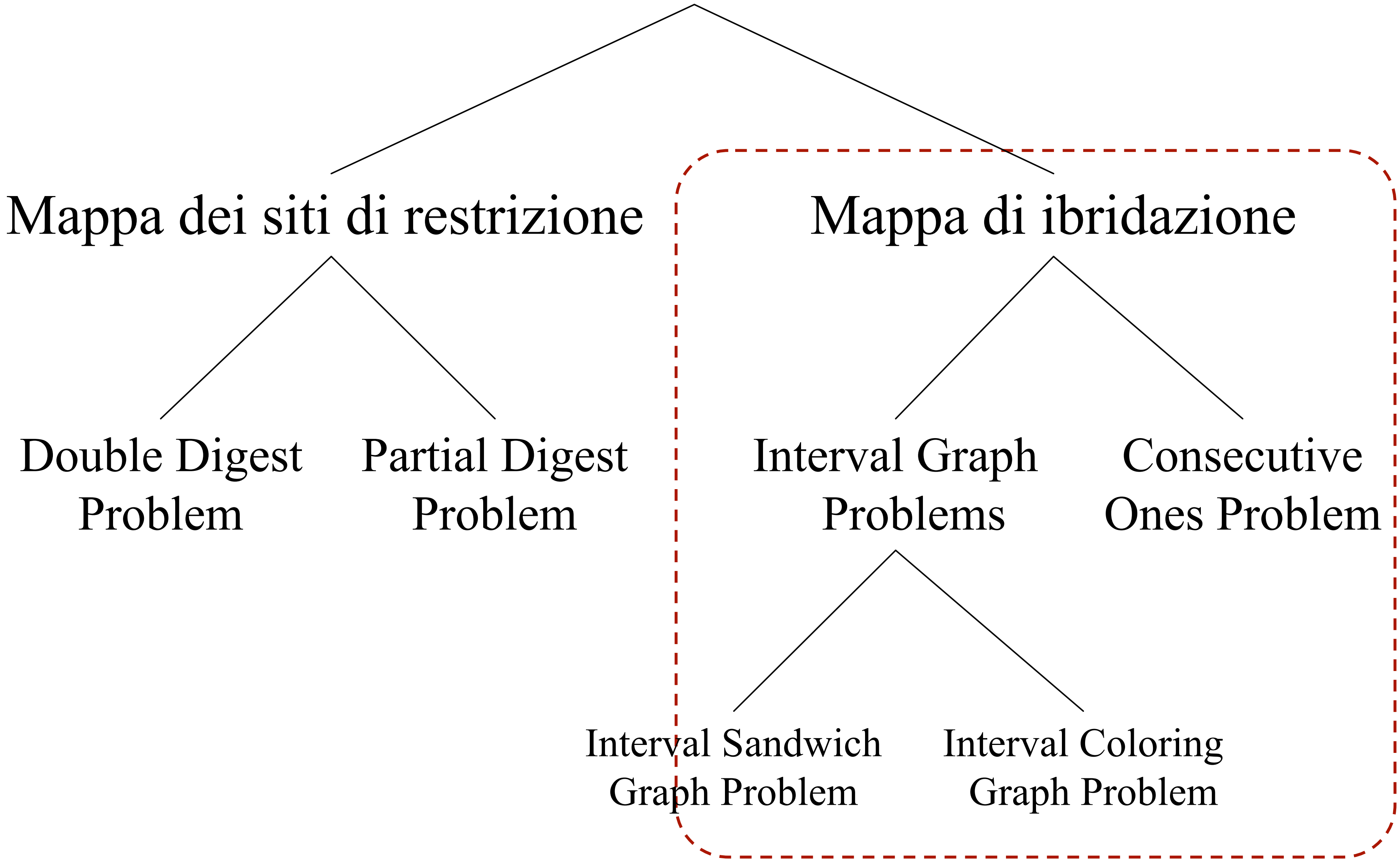
Mappa che individua la posizione di brevi sequenze note (marcatori) all'interno di una molecola di DNA (target DNA)

Fasi della costruzione di una *physical map*:
1. clonazione del target DNA
2. frammentazione
3. **ricostruzione dell'ordine reciproco dei frammenti lungo il target DNA**



Motivazione:
È impossibile ricostruire in laboratorio la sequenza delle coppie di basi di frammenti di DNA di lunghezza significativa

Physical Mapping



Consecutive Ones Problem

Costruire, a partire dai dati di un esperimento di ibridazione, una matrice binaria $A \in M_{(n,m)}\{0,1\}$ tale che

$A_{ij} = 1$, se la prova j ibrida il clone i, $A_{ij} = 0$ altrimenti.

Costruire, se esiste, una permutazione (C1P) delle colonne di A che renda consecutivi, sulle righe, gli elementi pari ad 1.

Una matrice binaria A soddisfa la *consecutive ones property* se ammette una permutazione C1P.

Assumendo che:
– l'esperimento sia privo di errori;
– le informazioni siano complete;
– le prove siano *uniche.*

$\Rightarrow$ La matrice A soddisfa la *consecutive ones property*

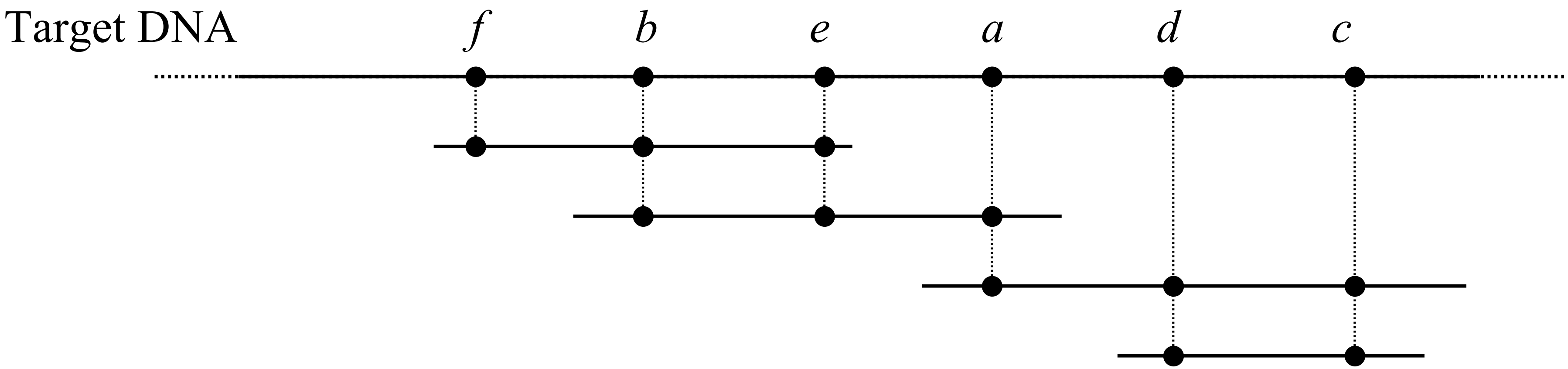
Algoritmo C1P

Input: Matrice A cloni per prove

Output: Matrice $\tilde{A}$, permutata secondo una permutazione C1P

$$A = \begin{array}{c} \\ \\ \end{array}\begin{array}{cccccc} a & b & c & d & e & f \end{array}$$
$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}$$

$$\begin{array}{cccccc} f & b & e & a & d & c \end{array}$$
$$\tilde{A} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$



Algoritmo C1P

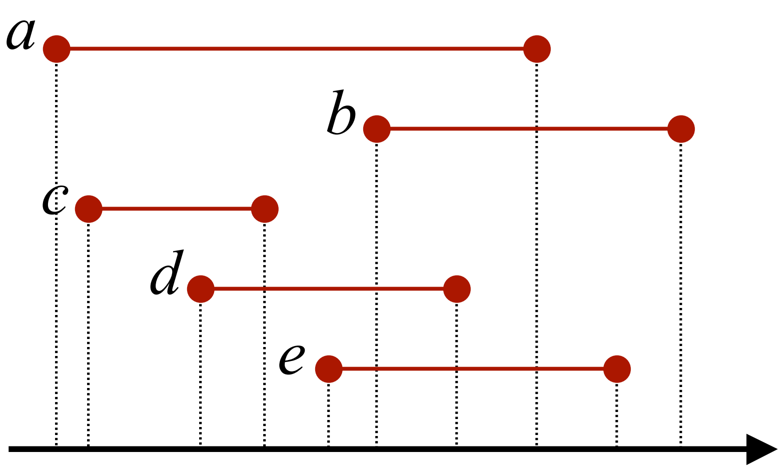
Input: Matrice A cloni per prove
Output: Matrice $\tilde{A}$, permutata secondo una permutazione C1P

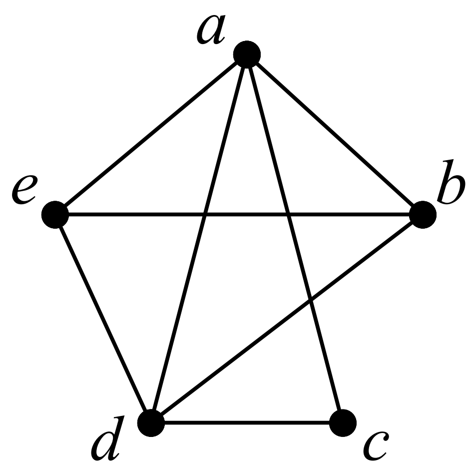
1. Trasformazione della matrice A nel grafo G “delle righe”
2. Costruzione del grafo G_C delle componenti connesse di G
3. Sort topologico di $V(G_C)$
4. Permutazione degli elementi relativi a ciascuna componente di G_C
5. L’unione delle singole permutazioni dà luogo alla matrice $\tilde{A}$ con la proprietà C1P

Implementazione dei passi dell’algoritmo e codifica in linguaggio C: complessità polinomiale $\boldsymbol{O(n^3m + n^2m^2)}$

Grafi intervallo

Un grafo non orientato $G = (V,E)$ è un ***grafo intervallo*** se $\exists\ f : V \rightarrow I$ tale che due intervalli si intersechino se e solo se i vertici corrispondenti sono adiacenti in G





Un grafo $G = (V,E)$, costruito sui dati di un esperimento di ibridazione, tale che:

$V = \{$ insieme dei cloni $\}$

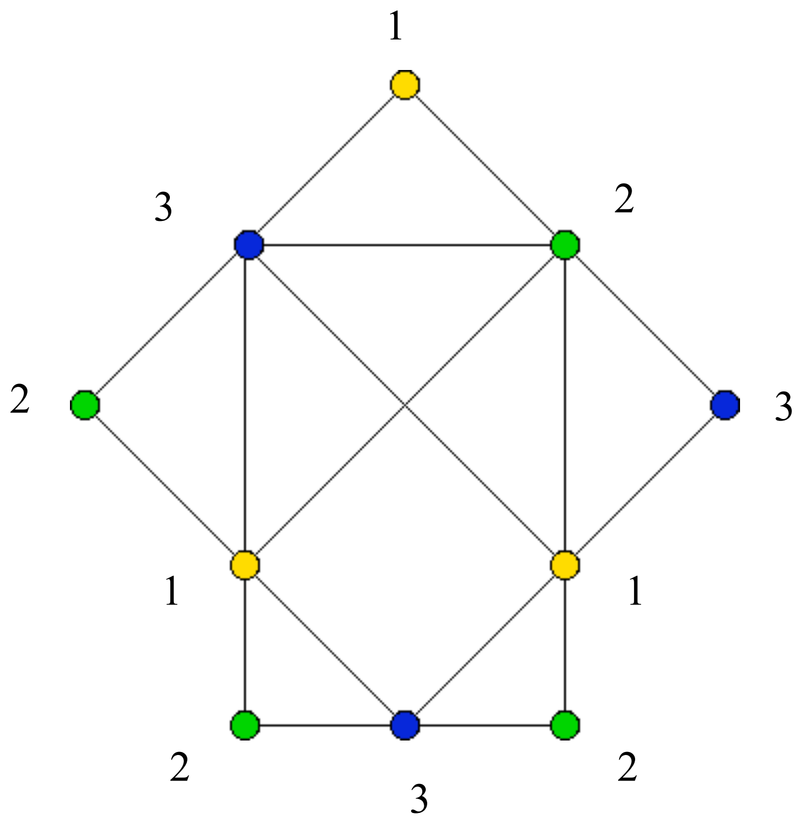
$E = \{ (u,v) \mid u,v \in V$ e i cloni u,v si sovrappongono $\}$

è un grafo intervallo.

Colorazione di un grafo

Un'applicazione $c : V \to \{1, 2, \ldots, k\}$ è una ***buona k-colorazione*** di G se due vertici adiacenti non hanno mai lo stesso colore:

$$(u,v) \in E \;\Rightarrow\; c(u) \neq c(v).$$



Problemi di ottimizzazione su grafi

Sandwich Graph Problem per la proprietà π

Dati due grafi $G = (V, E)$ e $G_F = (V, F)$ tali che $E \cap F = \emptyset$

Esiste un grafo $\hat{G} = (V, \hat{E})$ tale che $E \subseteq \hat{E}$, $\hat{E} \cap F = \emptyset$ e che soddisfa la proprietà π ?

Coloring Graph Problem per la proprietà π

Dato un grafo $G = (V, E)$ e una buona colorazione $c : V \rightarrow \mathbf{N}$
G è un sottografo di un grafo $\hat{G} = (V, \hat{E})$ per il quale c è una buona colorazione e che soddisfi la proprietà π ?

Il *coloring graph problem* è un caso particolare del *sandwich graph problem*, nel quale $F = \{ (u,v) \mid c(u) = c(v) \}$.

Errori nei dati biologici sperimentali

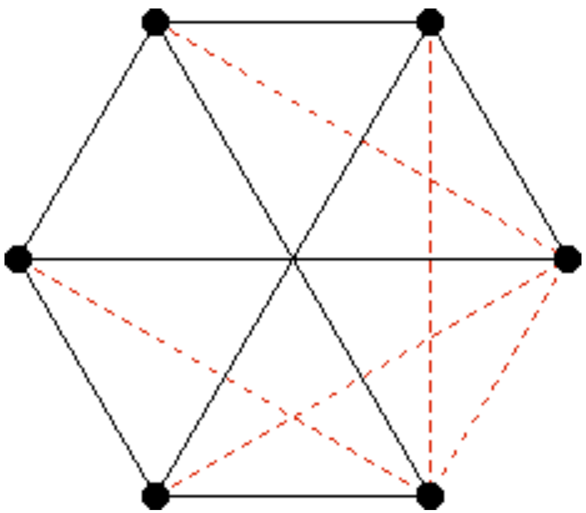
• ***falso positivo***: una prova ibrida un clone al quale non si sarebbe dovuta legare.
• ***falso negativo***: una prova non ibrida un clone al quale si sarebbe dovuta legare.
• ***clone chimerico***: due frammenti si uniscono durante il processo di clonazione e vengono replicati come clone unico.

Per questi motivi distinguiamo **tre tipi di coppie di cloni**:
– coppie di cloni la cui **sovrapposizione è certa** (insieme E);
– coppie di cloni la cui **sovrapposizione non è certa**;
– coppie di cloni che **certamente non si sovrappongono** (insieme F).

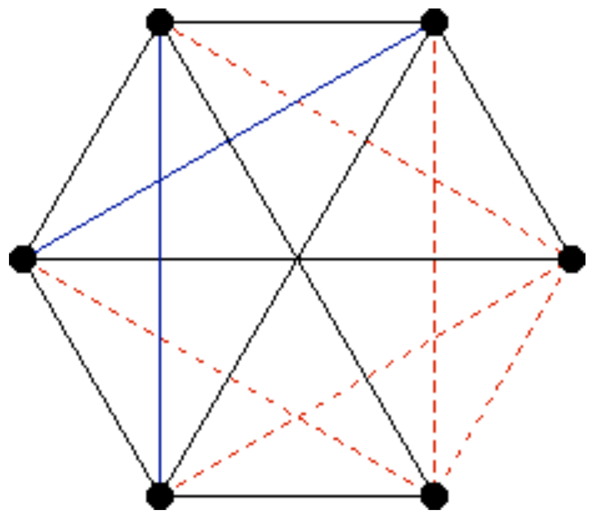
Rappresentando queste informazioni con un grafo possiamo tradurre il problema in una istanza di un *sandwich graph problem* $S = (V, E, F)$

Interval Sandwich Graph Problem

• ***Istanza***: $S = (V, E, F)$, con V un insieme di vertici ed E, F due insiemi disgiunti di spigoli.
• ***Problema***: Costruire, se esiste, un grafo intervallo $\hat{G} = (V, \hat{E})$ tale che $E \subseteq \hat{E}$, $\hat{E} \cap F = \emptyset$.



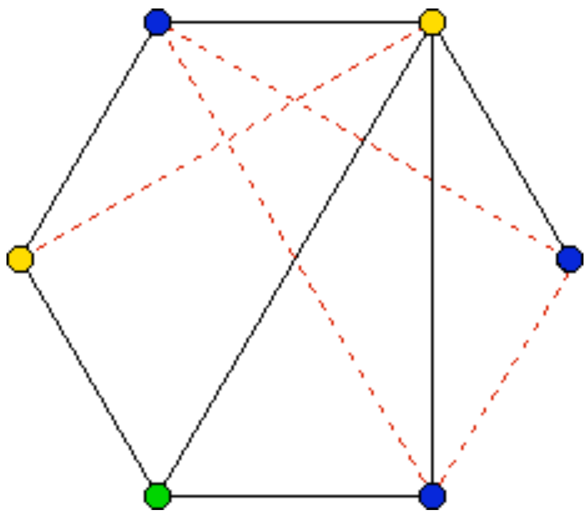
$S = (V, E, F)$



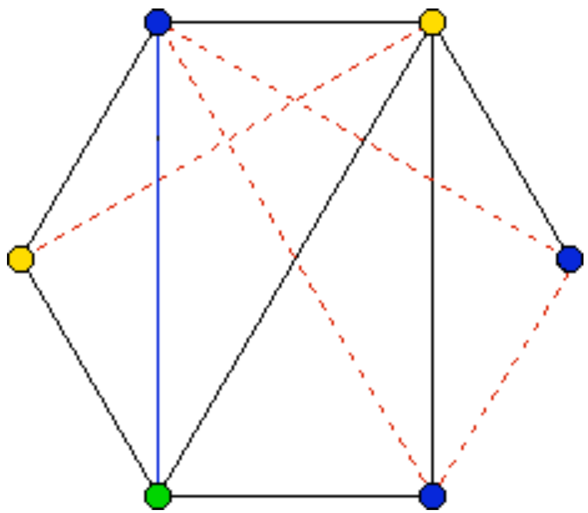
$\hat{G} = (V, \hat{E})$

Interval Coloring Graph Problem

- ***Istanza***: Un grafo $G = (V, E)$ e una buona colorazione $c : V \rightarrow \mathbf{N}$
- ***Problema***: Determinare se G è un sottografo di un grafo intervallo $\hat{G} = (V, \hat{E})$ per il quale c è una buona colorazione



$G = (V, E)$



$\hat{G} = (V, \hat{E})$

NP-completezza e parametrizzazioni

L'interval sandwich problem e l'interval coloring problem sono problemi **NP-completi**

Osservando meglio il problema biologico è evidente che alcuni "dati" possono essere interpretati come dei parametri del problema, poco rilevanti dal punto di vista della complessità

Si giunge così a versioni parametriche dello stesso problema, di complessità polinomiale:

- Limitazione del **grado** (d) dei vertici del grafo di input G, limitazione della dimensione (k) della ***clique*** massima del grafo intervallo $\hat{G}$

- Limitazione del **grado** (d) dei vertici del grafo intervallo $\hat{G}$

Parametrizzazioni

Limitazione del grado (*d*) dei vertici del grafo di input G, limitazione della dimensione (*k*) della clique massima del grafo intervallo Ĝ.	$\Leftrightarrow$	Limitazione, nei dati, del massimo numero di cloni che intersecano lo stesso clone; limitazione del massimo numero di cloni che si sovrappongono mutuamente nella mappa.
Limitazione del grado (*d*) dei vertici del grafo intervallo Ĝ.	$\Leftrightarrow$	Limitazione, nella mappa soluzione del problema, del numero massimo di cloni che può intersecare lo stesso clone.

Algoritmi

- Algoritmo "**SANDWICH**" esponenziale
- Algoritmo "**Parametric SANDWICH 1**" $O(n^{k\text{-}1})$
- Algoritmo "**Parametric SANDWICH 2**" $O(n^{d\text{-}1})$