UNIVERSITÀ DEGLI STUDI DI PISA

Facoltà di Scienze Matematiche Fisiche e Naturali
Corso di Laurea Specialistica in Informatica
Anno accademico 2006/2007

Tesi di Laurea

Scoperta di pattern ripetuti mediante l'uso di Alberi PQ

Candidato:
Giuseppe Camposeo

Relatori:

Prof. Roberto Grossi
Dott. Nadia Pisanti

Controrelatore:

Dott. Roberto Marangoni

A mia Madre.

Il sequenziamento dei genomi di molti organismi ha portato l'accumularsi di un enorme quantità di dati biologici da analizzarsi. Il problema maggiore consiste nell'estrarre, da questi dati, informazioni rilevanti dal punto di vista biologico.

Scopo di questa tesi è riuscire a fornire una nuova metodologia, che porti all'identificazione, all'interno delle sequenze genomiche, di particolari zone, con determinate proprietà combinatorie, tale da renderle uniche.

Partendo da un classico studio delle ripetizioni di permutazioni nelle sequenze biologiche, si è arrivati ad utilizzare la tecnica innovativa della struttura dati degli alberi PQ, che ha messo in evidenza le proprietà permutative delle sequenze genomiche analizzate. I test effettuati su reali sequenze biologiche, hanno messo in luce dei risultati sorprendenti, riuscendo ad individuare alcune zone, all'interno delle sequenze, dalle caratteristiche proprietà combinatorie.

Indice

CAPITOLO 1

Basi biologiche per l'analisi di sequenze genomiche

In questo capitolo verranno introdotte in maniera esemplificativa le basi biologiche che guidano gli approcci algoritmici per l'analisi di sequenze genomiche, in particolare quelli di tipo *discovery*, ovvero orientati all'identificazione di sequenze interessanti per qualche determinato criterio deciso a priori.

1.1 Nozioni biologiche

Il DNA (acido deossiribonucleico) è un *polimero*[1] organico costituito da *monomeri*[2], dette *basi*:

- *Adenina* (A)
- *Citosina* (C)
- *Guanin*a (G)
- *Timina* (T).

Timina e citosina sono dette basi *pirimidine*, mentre le altre due, guanina ed adenina sono dette *purine.* A ciascuna base è attaccato uno zucchero, *deossiribosio*, ed un gruppo fosfato. Tale gruppo chimico, cosi costituito prende il nome di *nucleotide*.

[1] Un polimero (dal greco *molte parti*) è una macromolecola, ovvero una molecola dall'elevato peso molecolare, costituita da un gran numero di piccole molecole (i *monomeri*) uguali o diverse (copolimeri) unite a catena mediante la ripetizione dello stesso tipo di legame.
[2] Col termine monomero (dal greco *una parte*) in chimica si definisce una molecola semplice dotata di gruppi funzionali tali per cui sia in grado di combinarsi ricorsivamente con altre molecole - identiche a sé o relativamente complementari a sé - a formare macromolecole.

L'unione tra nucleotidi consecutivi è di tipo covalente[3], ed avviene tra il fosfato del nucleotide precedente e lo zucchero del successivo, permettendo di realizzare filamenti di notevole lunghezza, più di un milione di nucleotidi consecutivamente attaccati.

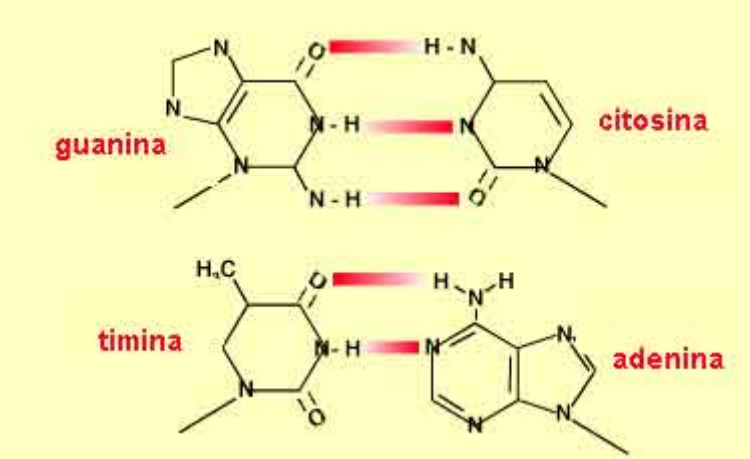

Figura 1

Nonostante i termini base e nucleotide rappresentano chimicamente cose diverse, nel linguaggio della biologia molecolare molto spesso si parla di basi intendendo i nucleotidi, dal momento che ciò che qualifica chimicamente il gruppo rimane sempre la base, poiché lo zucchero ed il fosfato sono uguali per tutti.

Il DNA rappresenta il materiale ereditario responsabile dell'informazione genetica della maggior parte degli organismi viventi. Nell'assoluta maggioranza dei casi, il DNA non si trova a filamento singolo, ma come una molecola a doppio filamento, con due filamenti appaiati antiparalleli. Tale appaiamento è reso possibile dalla complementarietà delle basi, cioè dalle proprietà chimico-fisiche che permettono alla base A di abbinarsi con la T, e alla C di abbinarsi con la G. Da un punto di vista

[3] Il legame covalente è un legame chimico di tipo forte.

chimico, tali appaiamenti sono legami di tipo idrogeno che risultano essere dei legami molto più deboli dei legami di tipo covalente.

1.2 Struttura del DNA

Una coppia complementare di filamenti di DNA assume una struttura a doppia elica (figura 2) del celebre modello presentato da James Watson e Francisi Crick (1953).

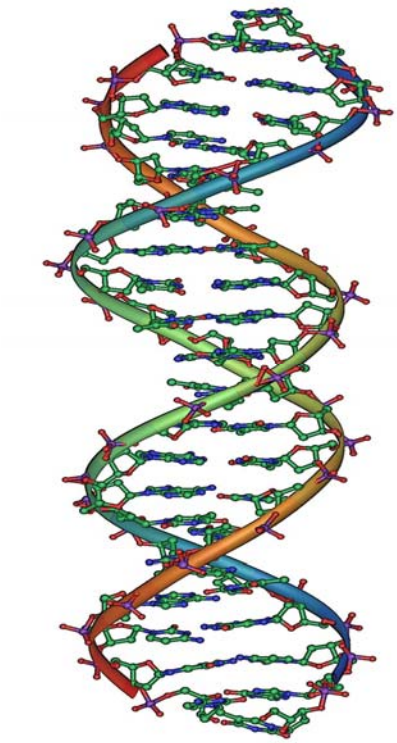

Figura 2

Le coppie di basi o paia di basi (dall'inglese *base pair*, bp o bps) sono comunemente utilizzate come misura della lunghezza fisica di una molecola di DNA.

Esistono vari tipi di conformazioni che la struttura del DNA può assumere:

- La forma *A* è un'ampia spirale destrorsa (doppia elica destrogira ma con 11 basi per giro d'elica), con un passo di 2,9 nm (circa 11bp) ed un diametro di 2,5 nm. Tale conformazione è presente in condizioni non fisiologiche, quando il DNA viene disidratato.
- La forma *B* è a doppia elica destrogira con 10 coppie di basi per giro dell'elica. Le interazioni idrofobiche fra basi consecutive sullo stesso filamento contribuiscono alla curvatura dell'elica.
- La forma *Z* è tipica invece delle sequenze che presentano modificazioni chimiche come la metilazione[4], e dei tratti di DNA ricchi di basi C e G. Essa assume un andamento sinistrorso, opposto rispetto alla conformazione B.

La rappresentazione classica del modello di DNA di Watson e Crick in forma di stringa richiede di usare due righe: nella prima si scrivono consecutivamente le lettere che rappresentano le basi di un filamento, mentre nella riga sottostante si trascrive il filamento complementare. Ad esempio:

. . . ATCCGT . . .
. . . TAGGCA . . .

Nella pratica si tende spesso ad eliminare la rappresentazione del filamento complementare, ritenendolo ridondante, ma si ha spesso la conseguenza che il bioinformatico dimentichi le implicazioni che la molecola presenta, proprio per il suo essere a doppio filamento. L'importanza di questo aspetto diverrà evidente illustrando la duttilità del DNA, le molteplici conformazioni che può assumere e le

[4] Il termine *metilazione* è usato in chimica per definire l'addizione o la sostituzione di un gruppo metile su vari substrati. Si tratta di un termine comunemente utilizzato in chimica, biochimica, e scienze biologiche.

proprietà di interazione con il macchinario molecolare della cellula che sono funzione della topologia locale di tratti di DNA.

La quasi totalità degli organismi viventi immagazzina l'informazione ereditaria in una o più molecole di DNA, dette cromosomi. I *procarioti*[5] posseggono un solo cromosoma, spesso di forma circolare, mentre gli *eucarioti*[6] posseggono molteplici cromosomi, di lunghezza variabile, sempre lineari. Negli eucarioti i cromosomi assumono forma diversa a seconda della fase del ciclo cellulare, e formano una massa indistinta nella cromatina nucleare durante la fase stazionaria, mentre durante la fase riproduttiva il nucleo scompare ed i cromosomi si impacchettano individualmente, assumendo la caratteristica forma ad "X", con le varianti "V" ed "Y". La fase interessante per questo studio di ricerca è quella stazionaria, dato che durante la riproduzione la normale attività trascrizionale viene sospesa. I filamenti che costituiscono i vari cromosomi sono mescolati insieme in modo apparentemente casuale, anche se misure molto recenti sembrano indicare che cromosomi diversi tendano ad occupare porzioni diverse del volume nucleare. L'organizzazione fine della cromatina vede il DNA avvolto intorno a proteine che lo costringono ad una superspiralizzazione che permette di compattarlo in modo altamente efficiente. Basti pensare che il genoma umano contiene un totale di 6×10^9 bp, e la sua dimensione lineare sarebbe compresa tra i 2.5 ed i 3 m, ed è completamente contenuto, insieme a molte altre molecole, in un nucleo cellulare dal raggio medio di 3 μm.

Ci sono principalmente due tecniche di indagine sperimentale di struttura 3D delle molecole biologiche: risonanza magnetica nucleare (NMR) e diffrazione a raggi X

[5] Le cellule procarioti sono tipiche degli archeobatteri, degli eubatteri e delle alghe azzurre. Esse sono relativamente piccole (con un diametro generalmente compreso tra 1 e 5 μm) e hanno una struttura interna alquanto semplice.

[6] Le cellule eucarioti costituiscono tutti gli altri organismi viventi (i protisti, le piante, i funghi e gli animali) sono molto più grandi (solitamente il loro asse maggiore è compreso tra i 10 e i 50 μm).

(X-Ray diffraction). Entrambe hanno un potere di discriminazione a livello atomico, ma moltissime limitazioni che ne impediscono l'impiego su molecole grandi. Di fatto quindi, non sono applicabili ad estensioni e complessità tipiche della cromatina nucleare. Comunque si possono osservare dei piccoli frammenti e dedurre conseguenze logiche da considerazioni generali di carattere chimico fisico. Ad esempio, tornando al concetto che C e G formano tre legami idrogeno, mentre A e T soltanto due, si comprende che zone del DNA molto ricche di GC siano relativamente *rigide* e difficili da *fondere*, cioè separarne i filamenti. A differenza di zone molto ricche di AT si prestano con più facilità a separare i filamenti e a dar vita a quelle strutture complesse (strutture a triplo o quadruplo filamento).

La capacità del DNA di assumere strutture tridimensionali particolari è di fondamentale importanza per le molteplici funzioni biologiche che esso svolge. Infatti, nonostante dal punto di vista strettamente chimico, ogni molecola di DNA sia sostanzialmente uniforme, le funzioni biologiche che possono trovare sede in specifiche regioni sono molteplici e assai diverse tra loro. Verranno presentate ora, alcune regioni funzionalmente diverse.

1.2.1 Regioni codificanti

Si chiamano *regioni codificanti*, i tratti che codificano le proteine la cui sintesi proteica si compone di due grandi fasi: la *trascrizione*, dove un tratto di DNA, appartenente ad un filamento definito, viene copiato in RNA[7], e la *traduzione*, dove

[7] L'acido ribonucleico (RNA) chimicamente è molto simile al DNA. Anch'esso è una catena polinucleotidica contenente quattro nucleotidi diversi. Le molecole di RNA differiscono da quelle di DNA perché contengono lo zucchero ribosio anziché il deossiribosio, una delle basi la timina è sostituita dall'uracile (U), e sono di solito a singolo filamento, anziché a filamento doppio.

l'RNA viene elaborato e successivamente tradotto in proteina. Entrambe le fasi, sono a loro volta composte da un gran numero di passaggi, realizzati mediante il coinvolgimento di proteine, enzimi e varie strutture cellulari. L'informazione passa dal DNA alla proteina secondo un codice, detto genetico. Tale codice, scoperto da Niremberg nel 1961 è sostanzialmente universale.

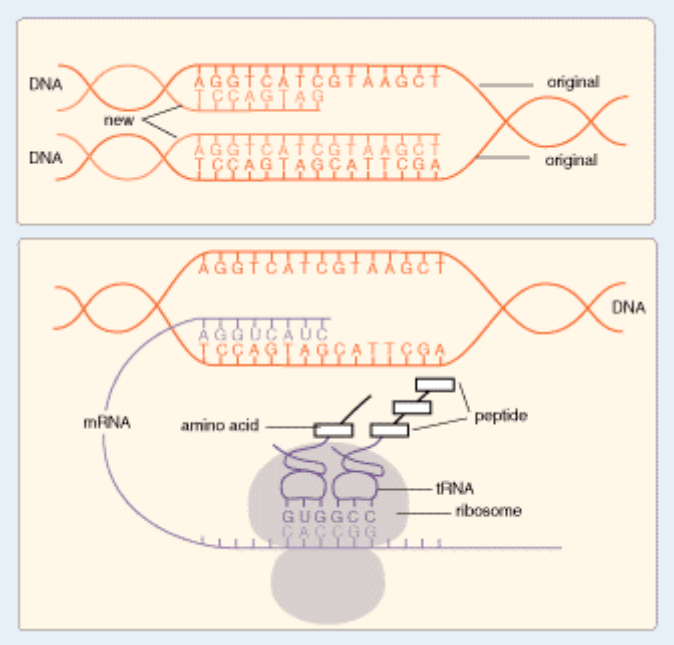

Figura 3: Interazione tra DNA ed RNA.

La regione del DNA che dà luogo ad una singola molecola di RNA trascritto viene detta *gene*. Occorre sottolineare come l'informazione che specifica la proteina sia custodita su uno solo dei due filamenti, mentre dal punto di vista del codice genetico, l'informazione contenuta sul filamento complementare è del tutto indipendente. Ad esempio, se la tripletta TAC specifica l'amminoacido *Metionina*, la sua complementare GTA specifica l'amminoacido *Istidina*. Ovvero, traducendo l'informazione dal filamento complementare si otterrebbe una proteina completamente diversa da quella normalmente specificata.

Negli eucarioti i geni sono interrotti, ovvero, può capitare che tra due zone che vengano tradotte in proteina si interponga una zona che non viene tradotta. Le zone tradotte si chiamano *esoni* mentre quelle non tradotte si chiamano *introni*.

Dal punto di vista informatico, le regioni codificanti si caratterizzano per essere più o meno casuali (distribuzioni quasi uniforme delle 4 basi), con una leggera correlazione sulla terza base, dove massimizzare la stabilità strutturale. In caso di possibilità di scelta è preferita una base in grado di instaurare legami più forti, come C o G.

1.2.2 Regioni regolatrici

I geni presenti nel genoma non sono tutti attivi simultaneamente. Alcuni vengono attivati sono in alcune fasi temporali dipendenti dal ciclo di vita dell'organismo, ad esempio lo sviluppo embrionale, altri sono la risposta a determinati stimoli esterni. Questa attività di accensione/spegnimento viene denominata *regolazione dell'espressione genica*. I meccanismi che la regolano sono molteplici e qualitativamente diversi. Essi vanno da un uso differenziale dell'impacchettamento (le regioni molto compresse non sono accessibili, quindi i geni contenuti non sono espressi) ad un intervento attivo di specifiche proteine. Quest'ultimo caso è sicuramente il più interessante perché è quello su cui si basa la normale regolazione dell'attività cellulare. Uno schema generale del meccanismo di questo tipo di regolazione prevede l'esistenza di una regione, detta *promotore*, localizzata a monte del gene da trascrivere, sulla quale si legano delle proteine dette *fattori di trascrizione* (TF). Le esatte sequenze dove i TF si attaccano vengono denominate *siti*

di attacco per fattori di trascrizione (TFBS) Quando i TF si legano come TFBS, la struttura locale del DNA viene modificata ed il gene soggetto a regolazione diventa disponibile a venir trascritto.

Un aspetto molto importante di questo meccanismo è che il riconoscimento dei TFBS da parte dei TF non è solamente basato sulla particolare sequenza di basi della regione del TFBS, ma sulla struttura tridimensionale che essa assume. La struttura assunta dal DNA dipende dalle basi esistenti in quelle regioni. Un esempio è raffigurato in figura 4.

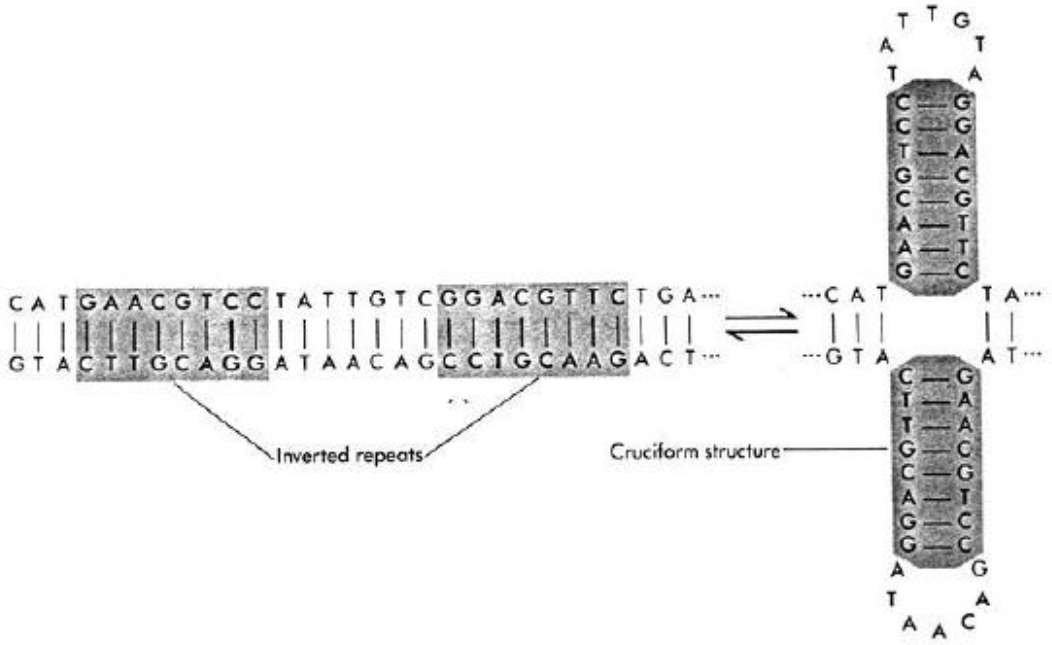

Figura 4

Quando due sequenze poste a distanza ravvicinata sono tali che l'una è la complementare inversa dell'altra (inverted repeats) la struttura 3D del DNA oscilla tra due forme in equilibrio: una di tipo lineare (la classica doppia elica) e l'altra di tipo cruciforme (a doppia forcina) che è strutturalmente molto complessa.

Organizzazioni simili a questa sono molto frequenti nel DNA e rappresentano il vero bersaglio di riconoscimento per i fattori proteici che al DNA si devono legare.
Ricerche che hanno usato approcci algoritmici orientati ad identificare tutte le zone in cui sono possibili simili organizzazioni strutturali, a prescindere dall'esatta sequenza di lettere che occorrono, ha portato all'individuazione, nelle regioni upstream di altri geni dello stesso organismo, di altre sequenze possibili candidati a TFBS.

1.2.3 Regioni ripetute

Le ripetizioni, dirette o invertite, hanno un'importanza fondamentale nello studio del genoma, perché ripetizioni di segmenti occorrono in una grande varietà di circostanze. Una classificazione che comprende alcune grandi sottoclassi vengono illustrate di seguito.

1.2.3.1 Ripetizioni nelle UTR del mRNA

Le ripetizioni nelle UTR (untranslated regions) del mRNA sono piccole ripetizioni di un singolo nucleotide, tipicamente poli A, a decine di ripetizioni di piccoli gruppi di lettere (coppie o triplette). Tali ripetizioni esercitano una regolazione del processo di traduzione del mRNA in proteina, influenzando quindi il tasso di produzione della proteina ed, in alcuni casi, anche la sua composizione.
Simili strutture ripetute figurano anche all'interno di introni, e sono spesso associate a sindromi di genetica dinamica, ovvero di patologie la cui insorgenza o intensità varia con il progredire delle generazioni. Tali patologie consistono in un ripetuto

aumento, o diminuzione, del numero di ripetizioni della tipologia in esame. Ad esempio se un soggetto normale mostra 11 ripetizioni di una coppia di lettere, e se ogni generazione aumenta di 3 il numero di ripetizioni, si avrà un nonno con 14 ripetizioni, un padre con 17 ed un figlio con 20. L'intensità e la rapidità di comparsa della patologia è proporzionale al numero di ripetizioni presenti, ed in questo caso, spesso abbiamo delle ripetizioni esatte.

1.2.3.2 Ripetizioni di geni

Geni molto simili, frequentemente, si presentano in cluster ravvicinati. Questo perché spesso geni simili derivano da duplicazioni di geni ancestrali e la localizzazione del nuovo gene è in tandem con il vecchio. Quindi si avrà che la duplicazione comprende tutta la struttura del gene, comprendendo esoni, introni ed eventuali sequenze regolatrici. Si tratta di ripetizioni con errore (o approssimate) dal momento che il processo di duplicazione non è esatto, e che la selezione naturale e l'evoluzione nel suo complesso favoriscono l'accumularsi di mutazioni che differenziano l'originale dalla copia.

1.2.3.3 Elementi mobili ed esogeni del DNA

Il DNA eucariota ospita al suo interno molteplici segmenti esogeni, appartenenti ad altri corredi genetici, e che sono integrati in tempi evolutivamente molto lontani. Tra loro, un ruolo di primaria importanza è svolto dai *trasposoni*, alcuni elementi genetici presenti nei cromosomi capaci di spostarsi da una posizione all'altra del

genoma, saltando su regioni diverse dello stesso cromosoma, o addirittura su altri cromosomi. I trasposoni, che vennero individuati inizialmente studiando le *cariossidi* (chicchi) del mais, sono presenti in tutti gli esseri viventi, sia in quelli più sviluppati, come l'uomo, sia nei batteri. Fanno parte degli elementi trasponibili, assieme alle *sequenze di inserzione*[8] IS. Come detto in precedenza, i trasposoni si spostano all'interno di uno stesso cromosoma oppure da un cromosoma ad un altro: per fare ciò hanno bisogno dell'enzima trasposasi, che viene codificato da geni presenti sui trasposoni stessi. I trasposoni presentano delle sequenze terminali invertite e si inseriscono in siti non omologhi, causando la formazione ai due lati di sequenze dirette ripetute, attraverso un taglio ineguale. Alcuni trasposoni (composti) presentano alle estremità alcune IS mentre altri hanno sequenze ripetute più corte; i trasposoni possono contenere anche geni per la resistenza ad un antibiotico e solitamente causano delle mutazioni, impedendo la trascrizione di un gene o ampliandola. Sono di diversa tipologia e mostrano lunghezze variabili da centinaia ad alcune migliaia di basi. Anche il loro numero di ripetizioni varia moltissimo, ma è comunque assai elevato e per alcune famiglie, può raggiungere le centinaia di migliaia. Ad una stima attuale, oltre il 30% del genoma umano, ed il 50% di alcune specie vegetali, sembra essere composto da elementi trasponibili di diversa lunghezza e numero di copie.

[8] In genetica, si definiscono *sequenze di inserzione* (denominate anche sequenze IS, elementi IS o, semplicemente, IS) delle sequenze di DNA capaci di spostarsi autonomamente da un punto all'altro del genoma.

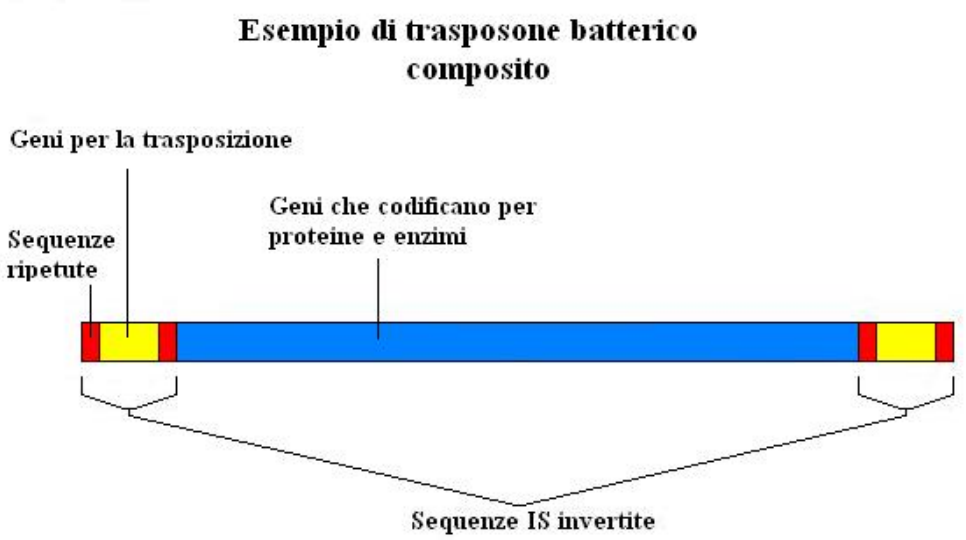

Figura 5

La figura 5, presenta alcuni trasposoni che con il meccanismo di taglia ed incolla riescono a rilocalizzarsi sul DNA. Si nota la presenza, agli estremi del trasposone, di coppie di sequenze complementate ed invertite, quindi, la ricerca di trasposoni, può tener conto sia di lunghe ripetizioni dirette, sia della compresenza di piccole ripetizioni complementate ed invertite.

1.2.3.4 Regioni di attacco alla matrice cromatinica

Scafford/matrix attachment regions (S/MARs) sono sequenze tipiche del DNA eucariota e corrispondono ai punti in cui il DNA si ancora alle proteine della cromatina e la loro localizzazione concorre alla regolazione dell'espressione genica. Le MAR vengono spesso identificate in vitro, quando frammenti di DNA vengono posti a contatto con proteine della matrice cromatinica e si osserva l'instaurarsi di un legame in una particolare zona del frammento. Si presuppone, per similitudine delle

condizioni chimico-fisiche, che la stessa cosa avvenga anche in vivo, all'interno del nucleo cellulare. Le S/MARs sono regioni di lunghezza variabile, da 100 bp a 1000 bp, non riferibili a vere sequenze consenso, quindi non identificabili coi metodi classici di pattern matching. Ad esempio, nell'uomo ne sono state stimate circa 300.000. Molte MAR, sono anche siti di origine di replicazione (ORI) ed è significativo come nell'uomo il numero delle MAR che sono anche ORI è stimato intorno ai 30 mila, che guarda caso, è anche il numero stimato di geni. Le MAR non contengono vere e proprie ripetizioni, ma piuttosto pattern di motivi variamente conservati, che vedono spesso una presenza molto alta delle basi A e T. Questo è in linea con la necessità che, in queste regioni, il DNA possa essere facilmente aperto e possa assume conformazioni peculiari. La struttura 3D non è stata ancora descritta, ma diverse realtà indicano che il DNA di una MAR può organizzarsi in *mispaired segments*, ovvero il punto dove l'appaiamento non è perfetto e si formano coppie usualmente non ammesse, tipo (A,C). Da un punto di vista di ricerca testuale, è molto probabile che sequenze MAR possano ospitare permutazioni esatte, o con errori di una determinata sequenza elementare.

1.2.3.5 Siti fragili e regioni altamente ripetute

Siti fragili e regioni altamente ripetute mostrano alcune analogie con le regioni MAR Al pari di esse sono spesso ricche di basi A e T, e mostrano un'organizzazione strutturale certamente complessa ed ancora ignota. I siti fragili, sono relativamente corti, meno di 100 bp, sono siti di frequente rottura del DNA, e le rotture che si

verificano durante la replicazione portano a gravi conseguenze, come cancellazione di geni o gruppi di geni, mutazioni cromosomiche, ecc.

Le regioni altamente ripetute, invece, sono regioni di lunghezza variabile (compresa tra i 10 bp ed i 1000 bp) e si ripetono un numero molto elevato di volte, ed anch'esse risultano prevalentemente ricche di A e T, e sono strutturalmente associate a due zone precise del cromosoma in fase riproduttiva: il *centromero*, ovvero nei cromosomi ad X, il centro dove le fibre del fuso cellulare si agganciano durante la fase riproduttiva per muovere i cromosomi, ed i *telomeri,* le estremità della X che hanno, tra le altre, la funzione di evitare l'incastro con altri cromosomi. In queste zone la ricchezza di A e T, unita alla frequente ripetizione, anche se non del tutto esatta, di motivi che possono anche essere palindromi (occorrenza diretta più complementata invertita) generano strutture 3D molto complesse. Anche in questo caso, una ricerca per segmenti permutati potrebbe dare grossi risultati e riuscire a unificare la descrizione del modulo unitario ripetuto, perché l'eventuale rumore puntiforme potrebbe invece rientrare nella descrizione a permutazione.

1.3 Indagini algoritmiche a "priori"

Il sequenziamento dei genomi di molti organismi (a settembre 2007, 639 completati e 3000 in corso di completamento) ha portato ad accumulare una quantità di dati impensabile solo fino a poco tempo fa. Uno dei principali e difficili problemi, consiste nell'estrarre, da questa enorme massa di dati, informazioni rilevanti ai fini biologici. Tale operazione non risulta affatto facile, perché le moderne tecniche di sequenziamento non producono alcuna informazione circa le proprietà biologiche

della regione sequenziata. D'altro canto, se sequenziare un genoma è divenuta un'operazione relativamente facile e veloce, acquisire informazioni sulle proprietà biologiche associabili alle varie regioni del genoma è una questione del tutto aperta, anche perché i meccanismi molecolari alla base di diverse funzioni cellulari sono, infatti, spesso ignoti, o noti solo in parte. Di conseguenza, la descrizione funzionale del genoma si arricchisce sempre più di particolari ed informazioni, anche qualitativamente nuovi, man mano che le ricerche procedono. Infatti, per questa ragione, i data base biologici sono in continuo aggiornamento, in cui la parte più variabile è rappresentata dalle *annotazioni*, ossia meta-dati che specificano informazioni su ruoli e proprietà svolti da determinate sequenze.

Grande è il numero di metodi sperimentali all'attribuzione di funzioni biologiche a sequenze genomiche, ma purtroppo davanti all'estensione di un genoma medio, mostrano un'inevitabile inadeguatezza, in particolare, richiedono tempi lunghi. Ed è questo il principale problema dell'era post-genomica, soprattutto per i ritardi che si creano tra l'accumulo dei dati "grezzi" e le acquisizioni di informazioni circa il ruolo biologico da loro ricoperto, che è poi l'aspetto veramente interessante della ricerca.

In questo contesto, l'adozione di approcci teorici, basati su assunzioni *a priori*, tipo la ricerca di tutte le ripetizioni approssimate di sequenze con lunghezza minima fissata e margine d'errore massimo fissato, o approcci teorici basati su *machine learning approaches*, ad esempio algoritmi adattivi che eseguono pattern recognition o clustering sulla base di un training set di casi noti, possono rivelarsi di grande aiuto per la ricerca genomica, permettendo un'analisi veloce di grandi masse di dati ed una selezione di sequenze candidate per l'attribuzione ad un ruolo specifico (ad esempio

coding/non coding, TFBS, S/MAR etc.) che possono rivelarsi di grande aiuto per la ricerca genomica.

Il problema risulta abbastanza complesso essenzialmente per due motivi. La prima causa è rappresentata dalla scarsa attendibilità dei criteri deducibili dalle ricerche genomiche, in base ai quali si dovrebbero impostare le ricerche a priori o mediante machine learning. Lo studio del genoma, infatti, risulta essere una scienza di recente sviluppo e la probabilità che future scoperte stravolgano impianti o criteri esistenti è molto alta. L'altra causa fondamentale risiede nella natura dei dati, dove i genomi, soprattutto quelli eucarioti, sono quanto più dissimile da una collezione di dati omogenei ed ordinati. Come è stato detto in precedenza, a dispetto dell'uniformità della chimica di base, il DNA è un groviglio di sequenze diversissime per struttura e funzione biologica.

Per spiegare meglio il problema di assegnare un ruolo alle sequenze genomiche, si può fare un'analogia molto efficace con il problema di comprendere cosa contenga un disco fisso di un computer sconosciuto. Ovvero paragonare una sequenza biologica ad un hard disk di un sistema sconosciuto. Ottenere una sequenza equivale ad ottenere un'immagine del disco, mentre comprendere la sequenza equivale ad un'ingegnerizzazione inversa del sistema, ovvero partire da un sistema sconosciuto e dedurne l'insieme delle specifiche dell'architettura e del sistema. [1].

Nonostante l'analogia sia stata pensata nel 1992, i termini del problema ed il contesto non sono variati di molto, ma comunque oggi disponiamo di un numero maggiore di dati, di criteri e di algoritmi per affrontare le ricerche.

I maggiori risultati, ottenuti con l'approccio a priori, sono stati ottenuti cercando all'interno dei genomi tratti ripetuti di lunghezza diversa e con diverso margine

d'errore, con il risultato di scoprire alcune sequenze regolatrici, i cluster di geni, il DNA altamente ripetuto, ed altre sequenze di significato biologico. Un approccio molto fruttuoso è stato quello di cercare pattern strutturati, che, come è stato visto in precedenza, può portare all'identificazione di TFBS ed altri elementi funzionali, come ad esempio i trasposoni, caratterizzati dal pattern a doppio palindromo alle estremità. La ricerca di permutazioni, invece, permetterebbe di individuare, con approcci a priori, altre sequenze di significato funzionale (S/MAR, altri tipi di trasposoni, etc.). Per sfruttare le potenzialità di questo tipo di ricerca, se ne deve definire un criterio per sequenze identiche a meno di permutazioni, eseguire una ricerca all'interno di una regione di genoma ricca di diverse funzionalità e annotata il più possibile in modo di avere la fortuna di trovare qualche specifica funzionalità in corrispondenza dei risultati della ricerca.

1.4 Stato dell'arte

Negli ultimi anni, l'analisi delle sequenze genomiche, ha avuto un ruolo sempre più importante nella ricerca scientifica, divenendo sempre più, un tema di grande attualità. Il numero di sequenze biologiche a disposizione è sempre maggiore, sommando le enormi dimensioni dei dati da analizzare, quindi, si ha il bisogno di avere strumenti in grado di identificare regioni del genoma ricco di particolari funzionalità. Un metodo *pattern discovery*, si pone l'obiettivo di individuare regioni simili all'interno di due o più sequenze per scoprire origini e funzioni evolutive comuni.

Nonostante l'analisi delle sequenze genomiche sia un campo di recente sviluppo, numerosi sono stati gli studi effettuati sul pattern discovery, di seguito un breve cenno:

- Un'applicazione chiamata GYM, basata su un metodo di *Data Mining* e *Knowledge Discovery* per trovare particolari pattern nelle sequenze proteiche [9].
- Un algoritmo che estrae da una sequenza di lunghezza n, tutti i pattern massimali di lunghezza k. L'algoritmo usa dei *Suffix-Tree*[9] per risolvere il problema in tempo computazionale $O(n\lg n)$. [10].
- L'Algoritmo *Teiresias* utilizzato per la ricerca di pattern in sequenze genomiche, utilizzando un metodo combinatorio per produrre tutti i pattern che compaiono in un numero minimo di sequenze, riuscendo ad essere molto efficiente evitando la completa enumerazione dell'intero spazio di ricerca del pattern. [11].
- L'algoritmo ToMMS permette di ricercare pattern frequenti da sequenze genomiche, basandosi sull'utilizzo di grafi, e ponendosi l'obiettivo di trovare un numero limitato di pattern, originando prima i pattern più specifici e poi quelli più generali. [12].
- Un algoritmo per la ricerca, all'interno di una sequenza, di pattern non ridondanti. L'idea di base è quella di partire dall'insieme di pattern non ridondanti per ricostruire tutti i pattern (ridondanti) presenti all'interno della sequenza. L'algoritmo prima individua tutti i pattern che hanno esattamente

[9] Un albero dei suffissi (*suffix tree* in inglese) è una struttura dati che evidenzia la struttura interna di una stringa in un modo che facilita operazioni comuni come la ricerca di sottostringhe. Gli alberi dei suffissi permettono di risolvere il problema del matching esatto in tempo lineare.

due caratteri specificati, all'inizio e alla fine, poi procede iterativamente concatenando due pattern ottenendone uno più lungo, in modo che l'insieme dei pattern si riduce drasticamente. [13].

- Un nuovo metodo per l'identificazione di pattern contenuti in un insieme di sequenze proteiche, non linearmente connesse. E' in grado di scoprire i pattern di una forma molto generale, distinguendoli per posizioni ambigue e per lunghezza variabile. [14].

CAPITOLO 2

Alberi PQ

In questo capitolo verrà presentata la struttura dati degli alberi PQ. Una struttura dati molto interessante ed utile per l'estrazione e la scoperta di pattern da sequenze biologiche. Verrà esposta la teoria che vi è alla base e che porterà ad enunciare l'algoritmo per la realizzazione di alberi PQ. Infine verrà analizzata brevemente la complessità del suddetto algoritmo.

2.1 Introduzione

Con il passare del tempo, è sorta l'esigenza di estrarre sempre più informazioni da dati di grossa dimensione. Uno dei campi in cui si è avuta questa necessità, in particolare quella di estrarre informazioni sempre più dettagliate da sequenze biologiche (stringhe), è la Biologia. Tale tema costituisce un importante punto di contatto tra la Biologia e l'Informatica, in quanto la computazione su stringhe costituisce un consolidato ambito di ricerca di notevole interesse in diversi settori dell'Informatica.

La computazione su stringhe in questo campo è un settore vasto che comprende diverse problematiche. Un tema centrale è lo studio di algoritmi su stringhe. In

passato ne sono stati sviluppati diversi, in particolare molti di questi rientrano in un settore noto come *pattern matching* comprendente differenti problemi che derivano dal ricercare particolari stringhe (pattern) in un testo.

In questo capitolo sarà esposta una nuova tecnica utilizzata per rappresentare e rivelare parti interessanti del genoma umano, ovvero la struttura dati degli alberi PQ. Tale metodo descrive la struttura più interna e le relazioni tra le varie parti di genoma, aiuta a filtrare pezzi di codice genetico apparentemente senza significato e costituisce, inoltre, il modo naturale per visualizzare parti di genoma strutturalmente complessi. In particolare, la struttura dati degli alberi PQ, aiuta ad ottenere facilmente la *massima notazione* tra un insieme di stringhe permutate tra loro. Più specificatamente, dato un insieme di stringhe permutate tra loro, la *massima notazione*, è un modo per rappresentare in maniera unica l'insieme sopracitato usando il carattere '**-**' tra un gruppo di uno o più geni che appaiono vicini tra loro in tutte le stringhe permutate. Ad esempio, date le due stringhe *A="abcde"* e *B="badce",* si ottiene che la loro massima notazione è uguale a *((a,b)-(c,d)-e)*.

2.2 Presentazione Alberi PQ

Gli alberi PQ (K. Both e G. Lueker 1976) sono una struttura dati ad albero utilizzati per rappresentare permutazioni su un insieme di elementi (stringhe). In particolare, Both e Leuker hanno realizzato gli alberi PQ per risolvere il seguente problema: dato un insieme finito X e una collezione *I* di sottoinsiemi di X, vedere se esiste una permutazione π di X dove ogni sottoinsieme *i* di *I* appare come una sottostringa consecutiva di π. Nel caso qui analizzato, data una stringa S (con $S=s_1s_2s_3\ldots s_n$ ed

$s_1 \ldots s_n$ appartenenti ad X ={A,C,G,T}) e un insieme di sue possibili permutazioni P = {p | p= π_i(S)}, l'idea è di trovare una rappresentazione unica dell'insieme P.

Both e Lueker hanno introdotto un efficiente algoritmo che risolva questo problema usando gli alberi PQ.

L'albero PQ è un albero con radice, i cui nodi interni possono essere di due tipi: P e Q. I figli di un nodo P possono occorrere in un ordine arbitrario (rappresenta la componente vera e propria della permutazione), mentre i figli di un nodo Q occorrono nell'ordine originale o, al massimo, in ordine inverso. Di norma, si usa rappresentare un nodo di tipo P con un cerchio (figura 6)

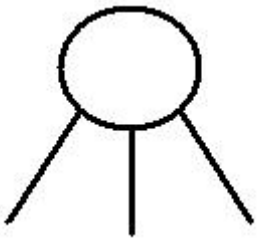

Figura 6

ed un nodo di tipo Q mediante rettangolo (figura 7).

Figura 7

Le foglie dell'albero sono etichettate dagli elementi di X (A,C,G,T nel caso di sequenze biologiche) e la frontiera[10] dell'albero è una stringa di elementi, permutati, di X.

Dati due alberi PQ T e T', si possono definire equivalenti (T ≡ T') se uno dei due è ottenuto dall'altro mediante una delle seguenti azioni:

1. Permutando arbitrariamente i figli di un nodo P.
2. Invertendo i figli di un nodo Q.

Ciò vuol dire che, definendo con C(T), l'insieme di tutte le possibili permutazioni della frontiera dell'albero T (F(T)), due alberi (T, T') sono equivalenti se C(T)=C(T') (figura 8).

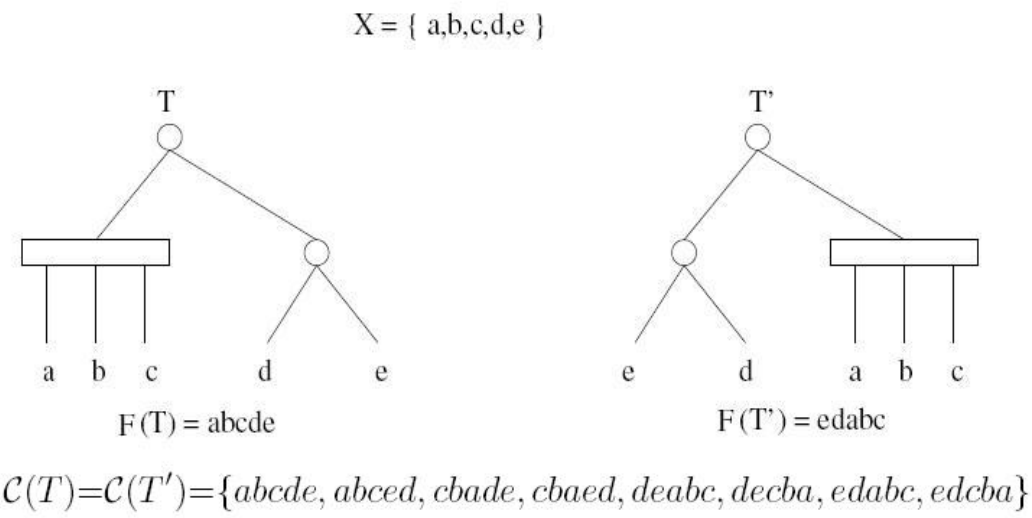

Figura 8

Si definiscono, infine, due particolari alberi PQ: dato un insieme finito X, l'albero formato da un solo nodo di tipo P con |X| figli, che sono tutte foglie, è chiamato *albero PQ universale*, indicato come T_U (figura 9).

[10] La frontiera di un albero T, denotata da F(T), è l'insieme delle foglie dell'albero, leggendole da sinistra a destra.

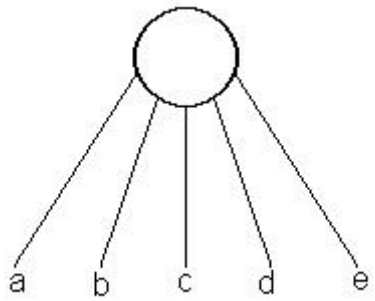

Figura 9

L'altro importante tipo di albero PQ è *l'albero nullo,* cioè un albero che non ha nodi.

2.3 Realizzazione degli alberi PQ

Alcune notazioni che saranno utili per il prosieguo del capitolo.

Innanzitutto si definisce con Π, l'insieme di stringhe permutate tra loro: $\Pi = \{S_1,S_2,S_3 \ldots S_k \mid S_1 = \pi_2(S_2) = \pi_3(S_3) = \ldots = \pi_k(S_k)\}$. Si assuma in più, una notazione scritta per definire gli alberi PQ. Questa notazione è ottenuta scrivendo l'albero PQ come una stringa *parentesizzata*, con differenti simboli per codificare un nodo P (usando la virgola come separatore "**,**") ed un nodo Q (usando come separatore il simbolo "**-**"). Ad esempio, l'albero T in figura 8, è definito come *((a-b-c),(d-e))*.

Quindi dato Π, il principale obiettivo è quello di costruire un albero PQ T a partire da Π, tale che la notazione scritta per definire l'albero PQ è uguale alla massima notazione di Π (definita nell'introduzione).

2.3.1 Minimo albero di consenso

Viene definito ora il *minimo albero PQ di consenso*: dato Π, T è un minimo albero PQ di consenso per Π, se $\Pi \subseteq C(T)$ e se non esiste $T' \neq T$ tale che $\Pi \subseteq C(T')$ e $|C(T')|<|C(T)|$.

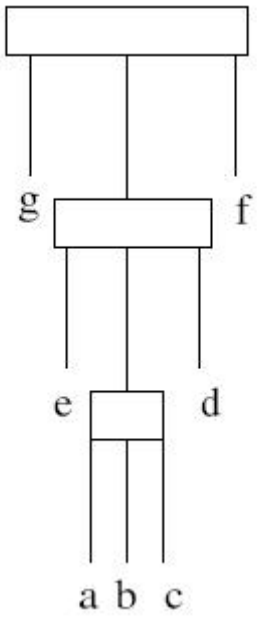

Figura 10:Dati π_1=geabcdf e π_2=fecbadg. L'albero in figura, T, è il minimo consensus PQ-Albero per l'insieme $\{\pi_1,\pi_2\}$.

Definito il minimo albero PQ di consenso, bisognerebbe ora trovare un metodo per costruirlo, ma prima è necessaria una premessa: l'algoritmo per la costruzione degli alberi PQ, come si vedrà in seguito, lavora su intervalli numerici. Le foglie dell'albero PQ, dunque, appartengono all'insieme dei numeri naturali N. Nel campo di applicazione delle sequenze biologiche, si avrebbero quindi dei problemi di attuazioni risolvibili facilmente mediante una semplice codifica. Ogni carattere delle stringhe del genoma umano è codificato con un numero intero distinto. Ci sono principalmente due modelli per codificare i singoli caratteri che rappresentano le

stringhe formanti il genoma umano, ovvero considerare la molteplicità o meno dei caratteri. Il caso di non considerare la molteplicità dei caratteri si rifà ad una banale codifica, considerando ogni singolo carattere l'uno diverso dall'altro, ed applicando una qualunque strategia nelle codifiche dei caratteri che si ripetono. Leggermente più complessa invece il modello di codifica in cui viene considerata la ripetitività dei caratteri. Ad esempio si considerino le seguenti due stringhe: *acbdefc* e *cdabfec*. Si nota come il carattere *c* si ripete più volte nelle due stringhe. Considerando la molteplicità, le due stringhe formeranno un albero PQ che corrisponderà alle stringhe *acbdefc'* e *cdabfec'* e considerando *c'* come un carattere distinto. Più specificatamente, si consideri il seguente insieme di permutazioni $\Pi_1=\{ p_1, p_2, p_3\}$ con le seguenti tre stringhe: p_1=*deabcxc*, p_2=*cdeabxc* e p_3=*cxcbaed*. Ogni carattere della prima stringa p_1 viene etichettato con un distinto numero intero, mentre le altre rimanenti stringhe vengono trattate come insiemi multipli. Avendo p_1=*deabcxc=1234567* verranno etichettate le altre due stringhe considerando la molteplicità del carattere *c* e quindi p_2=*cdeabxc=[57]12346[57]* e p_3=*cxcbaed=[57]6[57]4321*. Ovviamente, considerando la ripetitività, si avranno dei nuovi insiemi Π_i che porteranno alla costruzione di più alberi PQ. In figura 11, viene mostrato lo schema di tutte le possibili combinazioni di stringhe (quattro) che si vengono a creare considerando la ripetitività dei caratteri.

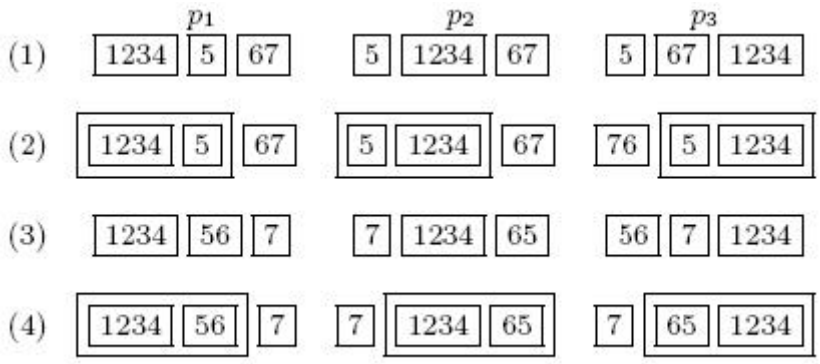

Figura 11

In figura 12 sono rappresentati gli alberi per i 4 casi sopracitati: $T_{1,3}$ rappresenta il primo e il terzo caso dell'esempio in figura 6, mentre T_2 e T_4 rappresentano gli alberi PQ per il secondo ed il quarto caso.

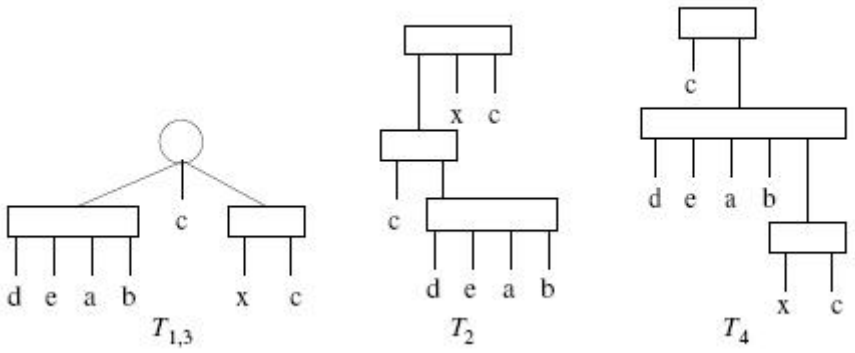

Figura 12

.

2.3.2 Intervalli comuni

Sarà utile definire il seguente insieme: l'*intervallo comune* di Π. Dato Π, si assuma $\pi_1 = id_n = (1,2,\ldots n)$. Un intervallo [i,j] ($1 \leq i < j \leq n$) è chiamato *intervallo comune* di Π (C_Π) se gli elementi dell'insieme {i,i+1,…..j} appaiono come sottostringhe consecutive in ogni π_n appartenente a Π (i=1,2,….k).

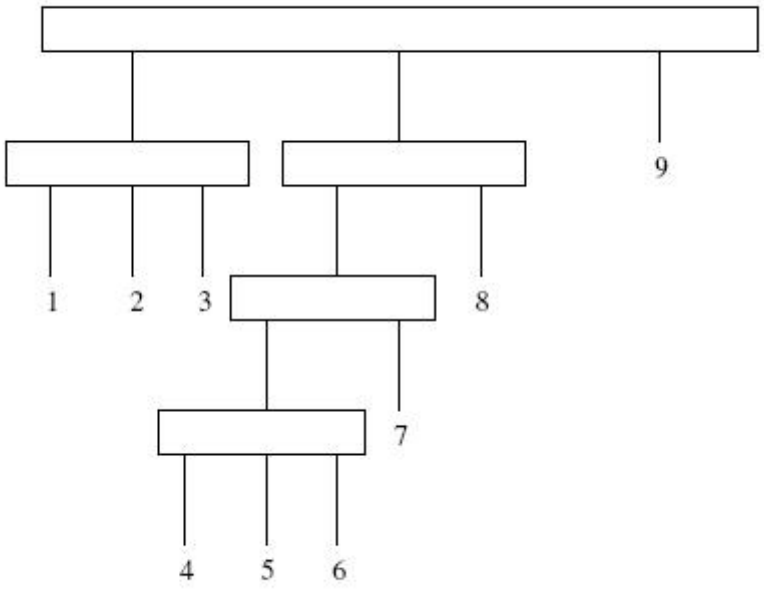

Figura 13

L'albero in Figura 13 è il minimo albero PQ di consenso per $\Pi = \{\pi_1,\pi_2,\pi_3\}$, dove π_1=(1,2,3,4,5,6,7,8,9), π_2=(9,8,4,5,6,7,1,2,3), e π_3=(1,2,3,8,7,4,5,6,9). L'intervallo comune per Π, C_Π = {[1,2],[1,3],[1,8],[1,9],[2,3],[4,5],[4,6],[4,7],[4,8],[4,9],[5,6]} mentre ((1-2-3)-(((4-5-6)-7)-8)-9) è la massima notazione.

Una funzione che farà da supporto nella costruzione dell'algoritmo che realizza un albero PQ, è la funzione *REDUCE*(I,T'): dato un insieme I formato da sottoinsiemi di N={1,2,…,n}, ed un albero PQ T', le quali foglie appartengono all'insieme {1,2,…,n}, la funzione *REDUCE*(I,T') costruisce un albero PQ T tale che, *f* appartiene a C(T) se *f* appartiene C(T'), e per ogni *i* appartenente ad I appare come una sottostringa consecutiva di *f*. La procedura ritornerà un albero vuoto se nessuna frontiera *f* appartiene a C(T'). Dalla definizione della funzione *REDUCE*(), si osservi che date due collezioni di sottoinsiemi di interi, I_1 e I_2, se $I_1 \subseteq I_2$ e T_1=*REDUCE*(I_1,T_U) e T_2=*REDUCE*(I_2,T_U) allora $C(T_2) \subseteq C(T_1)$.

Si può dimostrare che, dato l'insieme Π, T_C=*REDUCE*(C_Π,T_U) è un minimo albero PQ di consenso per l'insieme Π. Infatti, dalla definizione della procedura *REDUCE*(), sappiamo che $\Pi \subseteq C(T_C)$, quindi T_C è un albero PQ di consenso. Si può dimostrare inoltre che tale albero è anche il minimo. Si assuma che $T \neq T_C$ è un minimo albero PQ di consenso per Π, per cui esisterà una collezione I di sottoinsiemi di N tale che T = *REDUCE*(I,T_U). Ogni *i* appartenente ad I, appare come una sottostringa consecutiva per ogni frontiera *f* appartenente a C(T). Finché T è un minimo albero PQ di consenso per Π, ogni *i* appartenente ad I appare come una sottostringa consecutiva per ogni π appartenente a Π. Quindi ogni *i* appartenente ad I è un intervallo comune per Π e $I \subseteq C_\Pi$. Usando l'osservazione fatta in precedenza si ottiene che $C(T_C) \subseteq C(T)$.

2.3.3 Intervalli irriducibili

Per realizzare l'algoritmo che costruisce il minimo albero PQ di consenso in tempo computazionale lineare, bisogna trovare un sottoinsieme C_Π di grandezza *O(n)* che contiene sufficienti informazioni sulle k permutazioni. Considerando, ad esempio, il pattern {1,2,3} con le seguenti permutazioni Π = {123,321}, si avrà allora C_Π = {[1,2],[2,3],[1,3] }. Si è visto che il minimo albero PQ di consenso T è tale che per ogni *f* appartenente a C(T), gli insiemi {1,2}, {2,3} e {1,2,3} appaiono come delle sottostringhe consecutive. Si può osservare che l'intervallo comune [1,3] è ridondante, nel senso che se gli insiemi {1,2} e {2,3} appaiono come sottostringhe consecutive in ogni *f* appartenente a C(T), allora l'insieme {1,2,3} deve anche apparire come sottostringa consecutiva in ogni *f* appartenete a C(T). L'intervallo

comune [1,3], che è l'unione degli intervalli [1,2] e [2,3], non è pertanto necessario per la costruzione dell'albero T.

Si analizzano ora quali insiemi degli intervalli comuni sono necessari per costruire un albero T. Dato Π, si assuma che $\pi_1 = id_n = (1,2,\ldots n)$: i due intervalli comuni c_1 e c_2 appartenenti a C_Π hanno una non banale sovrapposizione se l'intersezione tra c_1 e c_2 è vuota ($c_1 \cap c_2 = \emptyset$) e non sono inclusi in ognuno di loro. Una lista $p = (c_1,c_2,\ldots,c_{l(p)})$ di intervalli comuni $c_1,c_2,\ldots,c_{l(p)}$ appartenenti a C_Π è una catena (di lunghezza $l_{(p)}$) se ogni due successivi intervalli nella lista p, hanno una non banale sovrapposizione. Un intervallo comune *I* è chiamato *riducibile* se c'è una catena non banale che lo genera, in altre parole *I* è l'unione di tutti gli elementi in tutti gli intervalli della catena, altrimenti è chiamato *irriducibile*. Questa trasformazione da intervalli comuni (C_Π) ad insiemi di intervalli riducibili ed insiemi di intervalli irriducibili, è indicata da I_Π, si avrà, quindi, che $1 \leq |I_\Pi| \leq |C_\Pi|$. Nell'albero dell'esempio in Figura 8, l'insieme irriducibile dell'intervallo comune di Π è: I_Π= {[1,2],[1,8],[2,3],[4,5],[4,7],[4,8],[4,9],[5,6] } e la loro catena è illustrata in figura 14.

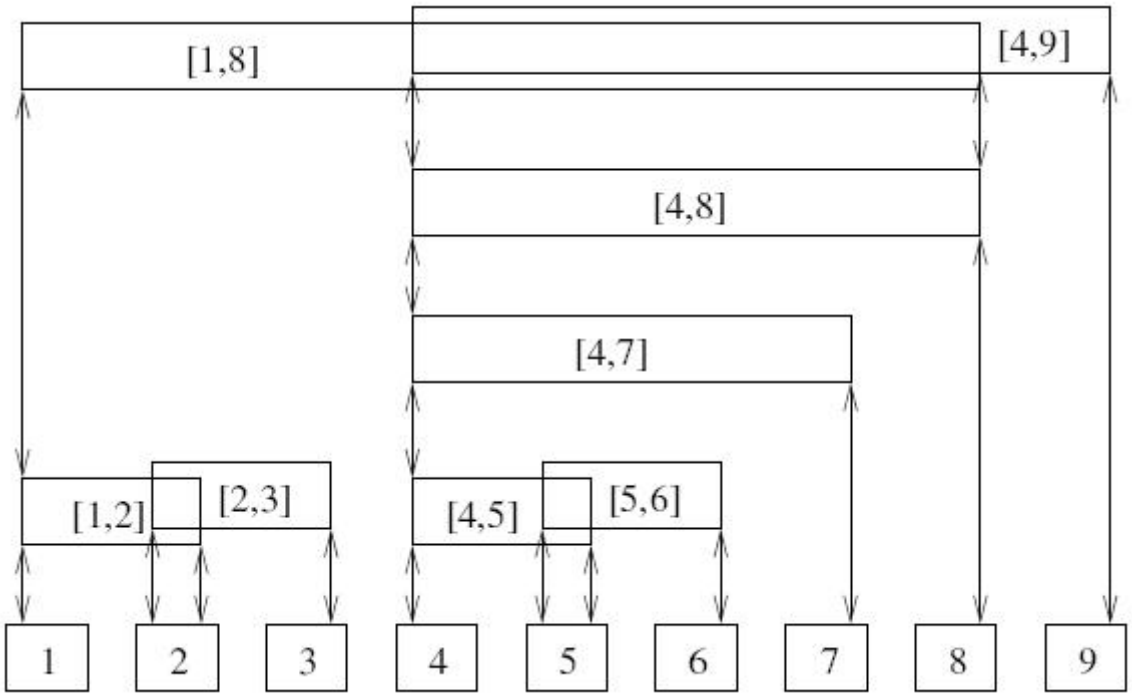

Figura 14

Le catene irriducibili con una non banale sovrapposizione dell'esempio in figura 9 sono: ([1,2],[2,3]), ([4,5],[5,6]) e ([1,8],[4,9]).

Verrà presentato ora un algoritmo che trova tutti gli intervalli irriducibili, dato l'insieme $\Pi=\{\pi_1,\pi_2\ldots..\pi_k\}$ con $k\geq 2$ permutazioni di interi $N=\{1\ldots..,n\}$, in tempo computazionale $O(kn)$.

L'idea di base dell'algoritmo è quella di costruire iterativamente l'intervallo irriducibile I_{Π} partendo da $I_{\Pi 1} = \{[j,j+1] \mid 1\leq j<n \}$. L'algoritmo calcola $I_{\Pi i}$ partendo da $I_{\Pi i-1}$ per i = 2…k. La funzione per costruire $I_{\Pi i}$ da $I_{\Pi i-1}$ sarà:

$$\varphi_i: I_{\Pi i-1} \rightarrow I_{\Pi i}$$

dove per ogni c appartenente all'intervallo $I_{\Pi i-1}$, $\varphi_i(c)$ è il più piccolo intervallo comune c' appartenete all'intervallo comune C_Π che contiene c. Poiché $I_{\Pi i} \subseteq C_{\Pi i} \subseteq C_{\Pi i-1}$ e sapendo che l'insieme degli intervalli comuni, che sono generati da sotto catene dell'intervallo irriducibile I_{Π}, equivale all'intervallo comune C_Π, $I_{\Pi i-1}$ genera

gli elementi di $C_{\Pi i-1}$, e $I_{\Pi i-1}$ genera anche $I_{\Pi i}$ .Si può facilmente vedere che *c'* appartiene all'intervallo irriducibile $I_{\Pi i}$ e che φ_i è una funzione surgettiva[11], cioè $I_{\Pi i} = \{\varphi_i(c) \mid c$ appartiene a $I_{\Pi i-1}\}$. L'algoritmo per calcolare gli intervalli irriducibili per k permutazioni è implementato dall'algoritmo 1.

Algoritmo 1 (Calcolo di I_{Π})

Input: Insieme $\Pi=\{\pi_1,\pi_2\ldots..\pi_k\}$ di k permutazioni di interi N={1…..,n}.

Output: I_{Π} .

1: $I_{\Pi 1} \leftarrow$ ([1,2],[2,3]……[n-1,n]).

2: **for** i=2,….k do

3: $I_{\Pi i} \leftarrow \{ \varphi_i(c) \mid$ c appartiene a $I_{\Pi i-1} \}$ // vedere Algoritmo 2.

4: **end for**.

5: output I_{Π} .

Sarà ora trattato come calcolare la funzione φ_i usata nell'algoritmo 1 e sarà rappresentato dall'algoritmo 2.

[11] Una funzione si dice surgettiva quando l'immagine coincide con il codominio, ovvero quando ogni elemento y del codominio è immagine di almeno un punto del dominio.

Algoritmo 2 (calcolo di $\varphi_i(I_{\Pi i-1})$)

Input: Due permutazioni $\pi_1 = id_n$ $\pi_2 = \pi_i^{-1}$; $I_{\Pi i-1}$.

Output: $I_{\Pi i}$

1: Inizializzazione Y e S.

2: **for** x = n-1,…..,1 **do**

3: Aggiornamento di Y e S

4: **while** ([x',y] ← *S.first_active_interval(x)*) definito **and** *f(x,y)* = 0 **do**

5: output [*l*(x,y),*u*(x,y)]

6: rimuovere [x',y] da la sua sottolista attiva.

7: **end while**

8: **end for**.

L'algoritmo è una versione modificata dell'algoritmo RC (*Reduce Candidate*) [12] dove le strutture dati Y (simile all'algoritmo RC) ed S (in RC non è presente, struttura che dipende da $I_{\Pi i-1}$) sono delle liste doppiamente linkate, con

$$f(x,y)=u(x,y)-l(x,y)-(y-x) \text{ dove}$$

$$l(x,y)=min\ \pi_2([x,y]) \text{ e}$$

$$u(x,y)=max\ \pi_2([x,y]).$$

Date in ingresso le permutazioni π_1 e π_i (π_i^{-1} rappresenta l'inverso della permutazione π_i), la struttura dati Y permette, per un dato x (ciclo *for,* passo 2), di accedere a tutti gli intervalli *non superflui*[13], y, di $C_{(\pi_1,\ \pi_i)}$, candidati a far parte degli intervalli irriducibili che si stanno costruendo. Lo scopo di S, invece, è ridurre ulteriormente

[12] Realizzato da T. Uno e M. Yagiura, è un algoritmo che trova tutti gli intervalli irriducibili, dati in ingresso due permutazioni. [15].

[13] Per un dato x, un intervallo [x,y] con y>x è detto *superfluo* se soddisfa f(x',y) > 0 per ogni x'≤x.

questi intervalli ai soli indici y, per i quali simultaneamente [x,y] appartenga a $C\pi_{i-1}$ (garantendo che [x,y] appartenga a $C_{\pi i}$) e che [x,y] contenga un intervallo *c* appartenente a $I_{\Pi i-1}$ che non è contenuto in nessun più piccolo intervallo di $C_{\pi i}$. Entrambe le strutture dati (Y e S) garantiscono che gli intervalli irriducibili vengano esattamente calcolati. La struttura dati Y è inizializzata con il solo elemento *n* mentre S è inizializzata partizionando $I_{\Pi i-1}$ in catene massimali con sovrapposizioni non banali. Per ciascuna di tali catene, S contiene una *clist* doppiamente linkata che inizialmente comprende gli intervalli della catena nell'ordine da sinistra a destra. Inoltre, intervalli di differenti *clist* con lo stesso inizio (estremo sinistro dell'intervallo) sono collegate mediante puntatori verticali. Per descrivere l'aggiornamento di S (linea 3), vengono introdotte le nozioni di *sleeping active* ed *intervalli satisfied*. Inizialmente tutti gli intervalli della lista *clist* sono definiti *sleeping*. In una generica iterazione x, tutti gli intervalli con l'inizio uguale ad x divengono *attivi* e sono inclusi alla testa di un, inizialmente vuoto, *active sublist*. Un intervallo rimane attivo finché è soddisfatto o cancellato. Per capire meglio l'aggiornamento di S si osservi la figura 15.

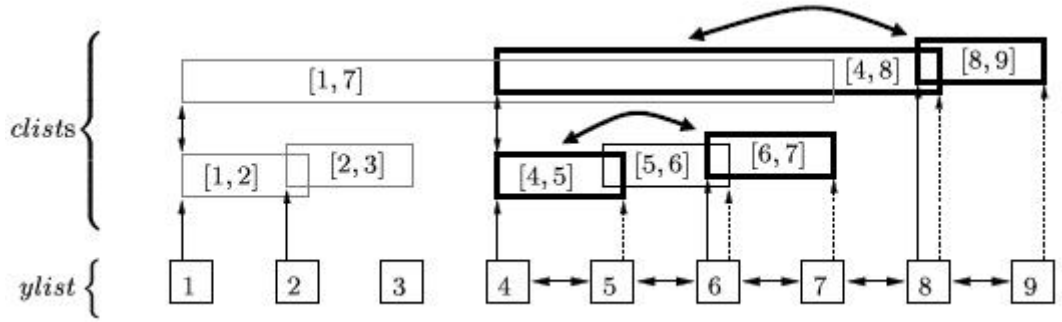

Figura 15

Essa descrive lo schema di *clist* e *ylis*t mentre viene elaborato l'elemento x = 4 di π_3 per l'insieme di permutazioni $\Pi=(\pi_1, \pi_2, \pi_3)$ analizzato nell'esempio di figura 13. I rettangoli ombreggiati rappresentano gli intervalli *sleeping*, quelli con linea spesse rappresentano gli intervalli *active* e quelli con linea sottile gli intervalli *satisfied*. Le frecce con linea di spessore collegano gli elementi di una sottolista *active*, le frecce verticali rappresentano le liste verticali (il puntatore verticale dell'indice 5 è stato cancellato dopo l'analisi dell'intervallo [5,6] nell'iterazione x=5), e le frecce tratteggiate verticali sono i puntatori di fine intervallo (estremo destro dell'intervallo). Infine la funzione *S.first_active_interval(x)* restituisce il primo intervallo *active* [x',y] contenuto in *clist* dell'intervallo della testa della lista verticale per l'indice x.

2.3.4 Algoritmo per la costruzione di alberi PQ

Il seguente algoritmo trae vantaggio dal fatto che gli intervalli irriducibili contengono tante informazioni quante sono quelle degli intervalli comuni. (C_Π).

Algoritmo 3 per la costruzione alberi PQ:

Input: Π (insieme di stringhe permutate tra loro).

Output: Albero PQ T.

1. Calcolare gli intervalli irriducibili I_Π.
2. Calcolare T=*REDUCE*(I_Π , T_U).
3. Ritorna T.

Si è già visto che T_C=*REDUCE*(C_Π,T_U) è il minimo albero PQ di consenso per un insieme di permutazioni Π. Si può facilmente dimostrare che se T=*REDUCE*(I_Π,T_U) allora T ≡ T_C, cioè il passo due dell'algoritmo 3 produce il minimo albero PQ di consenso.

L'algoritmo che permette la costruzione di alberi PQ dato un insieme di *k* permutazioni, ognuna lunga *n,* ha un costo computazionale pari a $O(kn+n^2)$. Questo perché gli intervalli irriducibili I_Π, ovvero l'algoritmo 1, sono calcolati in tempo $O(kn)$, e T=*REDUCE*(I_Π,T_U) ha un costo computazionale di $O(n^2)$ [16]. Quindi il costo totale dell'algoritmo 3 è $O(kn+n^2)$..

2.3.5 Costruire Alberi PQ in tempo lineare

In questo paragrafo verrà modificato il passo 2 dell'algoritmo 3 in modo da avere un costo computazionale lineare $O(n)$. Questo è possibile sfruttando la struttura dati S, introdotta nell'algoritmo per il calcolo degli intervalli irriducibili. Per ogni catena con una non banale sovrapposizione, S contiene una lista doppiamente linkata che contiene gli intervalli di questa catena nell'ordine sinistra-destra e che, inoltre, intervalli di differenti liste con lo stesso indice di sinistra o destra, sono collegati mediante puntatori verticali (figura 14). Verrà introdotta ora una nuova procedura chiamata *REPLACE*(S) che trasforma la struttura dati S in un minimo albero PQ di consenso. L'idea generale è quella di sostituire ogni catena con un nodo di tipo Q, dove i figli del nodo Q sono radici di sottoalberi con foglie prodotte dall'intersezione tra gli intervalli della catena. Ad esempio, in figura 14, la catena ([1,8],[4,9]) è sostituita da un nodo Q con tre figli, dove ogni figlio è radice di un sottoalbero con

foglie {1,2,3}, {4,5,6,7,8} e {9}. Ogni catena che non è una foglia o un nodo Q, e risulta puntato da un puntatore verticale, viene sostituita con un nodo di tipo P. Tonando all'esempio di figura 14, i link verticali da [4,8] a [4,7] ed 8 implicano che [4,8] è sostituito da un nodo di tipo P con due figli, dove ogni figlio è radice di un sottoalbero contenente rispettivamente le foglie {4,5,6,7} e {8}. Finalmente un nodo P con due figli è sostituito con un nodo Q. L'albero PQ ottenuto da *REPLACE*(S) è illustrato in figura 8.

Sapendo che I_{Π} e S possono essere elaborati in $O(kn)$, e che *REPLACE*(S) può essere implementata mediante una semplice visita trasversale bottom-up di S in tempo $O(n)$, il minimo albero PQ di consenso, pertanto, può essere costruito con un costo computazionale di $O(kn)$.

CAPITOLO 3

Un approccio di Data Mining su sequenze biologiche

In questo capitolo verrà descritta e discussa l'analisi ed i test effettuati in questa tesi sulle proprietà combinatorie di alcune sequenze biologiche, in particolare sulle permutazioni di basi contenute in segmenti biologici, utilizzando la struttura dati degli alberi PQ, descritta nel capitolo precedente, che hanno messo in evidenza alcune particolari zone dalle interessanti proprietà strutturali. Verrà discusso il modo con il quale si è arrivati ad usare gli alberi PQ, ed inoltre saranno presentati e commentati i risultati ottenuti.

3.1 Test preliminari

Lo scopo principale di questa tesi è quello fornire una nuova metodologia per analizzare alcune sequenze di DNA, cercando di individuare delle zone interessanti, presenti in essa, che potrebbero portare alla rilevazione di particolari funzionalità presenti nella sequenza genomica. Le sequenze che sono state prese in considerazione, sono sequenze genomiche, ortologhi[14] delle sequenze umane e

[14] Geni ortologhi sono geni simili riscontrabili in organismi correlati tra loro. Il fenomeno della speciazione (processo evolutivo) porta alla divergenza dei geni e quindi delle proteine che essi codificano.
Ad esempio l'α-globina di uomo e di topo hanno iniziato a divergere circa 80 milioni di anni fa, quando avvenne la divisione che dette vita ai primati e ai roditori. I due geni sono da considerarsi ortologhi.

topesche, in studio presso l'ospedale Gaslini[15] di Genova, che presentano al loro interno particolari funzionalità biologiche. L'obiettivo di questo capitolo, quindi, è di studiare le sequenze in questione, cercando, al loro interno, proprietà combinatorie tali da mettere in evidenza particolari zone con particolari *feature* biologiche. Il primo passo è stato quello di calcolare tutte le sottosequenze permutate, presenti all'interno della sequenza genomica, nella speranza che i risultati ottenuti portassero all'individuazione di particolari aree dei geni studiati, con determinate proprietà permutative.

3.1.1 Ricerca di permutazioni

Nella speranza di individuare particolari strutture, presenti all'interno delle sequenze biologiche, si è partiti cercando ed analizzando le possibili permutazioni[16] di basi presenti all'interno di segmenti genetici, con l'obiettivo di individuare qualche zona di particolare interesse biologico.

3.1.1.1 Permutazioni con sovrapposizioni

Il primo passo è stato quello di trovare tutte le ripetizioni di permutazioni di dimensione *n,* considerando anche le sovrapposizioni, dove per sovrapposizioni s'intende la possibilità di considerare stringhe permutate tra loro che presentono un

[15] L'Istituto Giannina Gaslini è un Ospedale Pediatrico. I suoi scopi istituzionali sono il ricovero e la cura dei pazienti in età pediatrica, la ricerca in campo biomedico, la formazione continua degli operatori sanitari.
[16] Date due stringhe A e B, definiamo $A=\pi(B)$ dove $\pi(B)$ rappresenta l'insieme delle permutazioni di B.

parziale accavallamento. Ovvero, data una sequenza T, due occorrenze di sottosequenze permutate T[i..j] e T[i'..j'] sono parzialmente sovrapposte se i' ≤ j.

Figura 16

Ad esempio (figura 16) in T[0..18]=*tgacttgcagtaccgtact* abbiamo tre occorrenze di permutazioni di dimensione *n=5*: T[2..6]=*acttg*, T[4..8]=*ttgca* e T[14..18]=*gtact*. Come si può notare, l'inizio della seconda sottosequenza si sovrappone con la prima sottosequenza trovata.

Quindi quello che è stato fatto, dato in ingresso una sequenza DNA, e di calcolare tutte le sottosequenze permutate tra loro di lunghezza *n*, ovvero, presa una sottosequenza S_1, di dimensione *n*, vengono trovate tutte le altre sottostringhe S_i permutate, presenti all'interno della sequenza biologica, creando così l'insieme $\Pi=\{S_i \mid S_1 = \pi_i(S_i)\ 1\leq i\leq t\ \}$, dove t sono le occorrenze di permutazioni trovate. Nell'esempio di figura 16, avremo l'insieme Π={ *acttg*, *ttgca, gtact* }. Ovviamente lo stesso procedimento è stato ripetuto per ogni sottosequenza, diversa da S_1 (sempre di dimensione *n*) contenuta nella sequenza analizzata, ottenendo l'insieme di insieme di permutazioni $\Pi=\{\Pi_1, \Pi_2, \ldots, \Pi_p\}$, dove ogni Π_j è del tipo spiegato in precedenza.

La prima prova, su una sequenza di DNA di topo, è stata effettuata con una finestra di dimensione *n*=200, ed in tabella 1 sono presentati i risultati.

n	Permutazioni trovate
200	363214

Tabella 1

Come si può notare, il numero di sottosequenze permutate trovate, di dimensione 200, risultano essere più di 300 mila (considerando che ogni insieme di permutazioni è formato da una quindicina di sottosequenze, abbiamo più di 25 mila insiemi di permutazioni). Questo mostra sorprendentemente che le permutazioni si ripetono più volte non consentendo né di individuare particolari regioni all'interno della sequenza, né di capire se tra queste permutazioni ve ne sono alcune di particolare interesse, visto l'elevato numero.

Per cercare di ottenere migliori risultati, si è ripetuta la prova, allargando la finestra di ricerca *n,* nella speranza di raffinare i risultati ottenuti, individuando, quindi, zone ricche di proprietà combinatorie.

n	Permutazioni trovate
300	345506
400	327767
800	279817

Tabella 2

Nella tabella 2, sono presentati i risultati ottenuti, variando la dimensione della finestra di ricerca *n* (300,400,800), ma come si può vedere, non si sono ottenuti gli esiti sperati, visto che, il numero di sottosequenze permutate trovate rimaneva sempre alto, nonostante diminuissero, rispetto alla prima prova, con l'aumentare della finestra *n*.

3.1.1.2 Permutazioni disgiunte

Per cercare di ridurre il numero di permutazioni trovate all'interno della sequenza DNA, si è deciso di non considerare le sovrapposizioni, discusse nel precedente paragrafo, e quindi considerare le sole permutazioni disgiunte presenti all'interno della sequenza biologica. In altri termini, data una sequenza T, due occorrenze di sottosequenze permutate T[i..j] e T[i'..j'] sono disgiunte se i'> j.

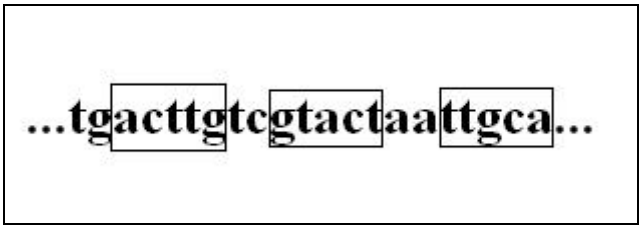

Figura 17

Ad esempio (figura 17) in T[0..20]=*tgacttgtcgtactaattgca* abbiamo tre occorrenze di permutazioni di dimensione *n=5*: T[2..6]=*acttg*, T[9..13]=*gtact* e T[16..20]= *ttgca.*

Come nel test precedente, sono state effettuate le ricerche di permutazioni, dato in ingresso una sequenza biologica, e variando la finestra di ricerca (dimensione della sottosequenza *n*).

Dimensione stringa *n*	Permutazioni trovate
200	298047
800	più di 200000

Tabella 3

I risultati, riportati in tabella 3, anche non considerando le sovrapposizioni, indicano un numero ancora elevato di permutazioni trovate non consentendo di fare nessuna supposizione sulle proprietà combinatore presenti all'interno della sequenza.

Il passo successivo sarebbe stato quello di cercare le permutazioni, considerando anche un certo margine d'errore, ma visto il numero elevato di permutazioni esatte trovate, si è deciso di non procedere, perché si avrebbero avuti dei risultati di dimensioni maggiori di quelli ottenuti nelle precedenti prove.

3.1.2 Catene massimali

I risultati ottenuti nei test precedenti, mostrano sorprendentemente che le sottosequenze permutate si ripetono più volte, non consentendo di capirne né la struttura né possibili zone biologicamente interessanti presenti all'interno del DNA. Quindi si è cercato di estendere il concetto di permutazioni considerando una nuova definizione di ricerca: dato la sequenza T ed una lunghezza *m*, una sua sottostringa T[i..j] è *m-massimale* se T[i..i+*m*-1] appare permutato varie volte in T[i..j] in modo da ricoprire tutte le posizioni in [i..j], e questa proprietà non vale per T[i-1..j] o T[i..j+1]. Ad esempio, in T[0..9] = *atagagaaca* abbiamo che T[2..7]=*agagaa* è massimale con m = 3 e T[i..i+m-1] = *aga*. Si noti che una sottostringa *m*-massimale è caratterizzata dai suoi primi *m* caratteri modulo le sue possibili permutazioni. Quindi, una ripetizione non è più una sottostringa di lunghezza *m* eventualmente permutata, ma sono due sottostringhe *m*-massimali (e possono avere anche lunghezze diverse) tali che i loro primi caratteri sono uguali se permutati opportunamente.

Si sono cercate, quindi, tutte le catene *m*-massimali (per *m* uguale a 200, 400, 800), presenti all'interno delle sequenze genomiche analizzate, ottenendo i risultati illustrati in tabella 4.

m	catene *m-massimali* trovate
200	47897
400	71668
800	84686

Tabella 4

Nonostante l'intuizione delle catene *m*-massimali, i risultati non sono stati quelli sperati, visto che per ogni *m*, l'enorme numero di catene trovate, non ci consente di individuare particolari zone all'interno delle sequenze genomiche analizzate.

3.1.3 Applicazione degli alberi PQ

Le prove effettuate precedentemente, hanno dimostrato l'enorme complessità combinatoria che le sequenze genomiche analizzate possiedono. Quindi servirebbe un metodo per verificare se tra le innumerevoli permutazioni ne esistono alcune dalla struttura particolare, ovvero trovare un modo per poterne studiarne le proprietà combinatorie. Per far ciò, viene utilizzata la struttura dati degli alberi PQ. Come già trattato nel capitolo precedente, gli alberi PQ descrivono la struttura più interna e le relazioni che ci sono tra un insieme di stringhe permutate, quindi, oltre a filtrare le stringhe permutate,ed aiutare ad evidenziare particolari zone del genoma umano, ne descrivono le strutture combinatorie presenti al loro interno.

Si considerino, ad esempio, due insiemi di permutazioni, ricavati da una sequenza di DNA: Π_1={*cggtatc, acggttc, atccggt*} e Π_2={*agaagct,gtagaac,acatgga*}. A prima vista i due insiemi sembrerebbero non dare alcuna informazione sulla loro struttura tale da poter individuare una particolare zona all'interno della sequenza genomica.

Come visto nel capitolo precedente, si costruiscono i due alberi PQ per i rispettivi insiemi,.

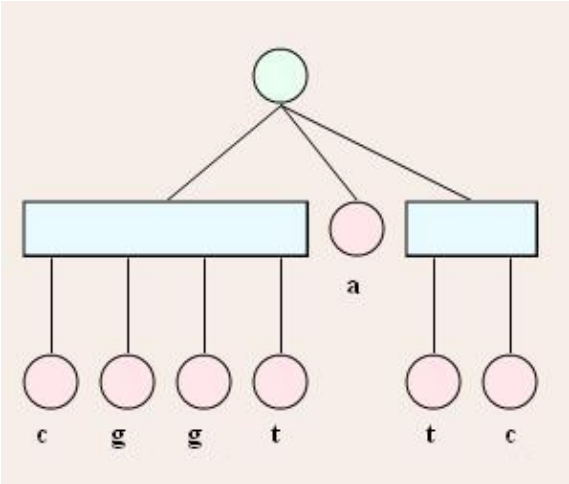

Figura 18: Albero PQ per l'insieme Π_1

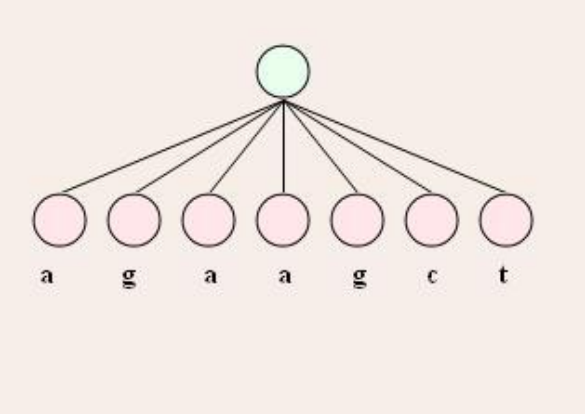

Figura 19: Albero PQ per l'insieme Π_2

E' chiaro che, guardando i due alberi, l'insieme di permutazioni Π_1 (figura 18) ha una struttura più complessa, rispetto all'insieme di permutazioni Π_2 (figura 19) che porta alla creazione di un albero con un solo nodo di tipo P come radice e 7 figli come foglie. Questo dimostra la non correlazione, sotto il profilo combinatorio, che c'è tra le stringhe che compongono l'insieme Π_2, a differenza delle stringhe che compongono l'insieme Π_1, dove l'albero PQ realizzato mostra una stretta

connessione tra le stringhe che lo producono (si noti, ad esempio, la vicinanza tra le basi *cggt* in tutte e tre le stringhe).

Si applicherà ora la teoria degli alberi PQ alle oltre 300 mila permutazioni trovate nelle sequenze di DNA, per vedere se è possibile effettuare una scrematura sull'insieme di permutazioni trovate.

3.2 Applicazione realizzata

Prima di passare a presentare ed analizzare i test effettuati utilizzando gli alberi PQ, viene data una breve descrizione dell'applicazione realizzata per effettuare le prove. E' stata realizzata un'applicazione che, data in ingresso una sequenza genomica, restituiva in output un insieme di alberi PQ.

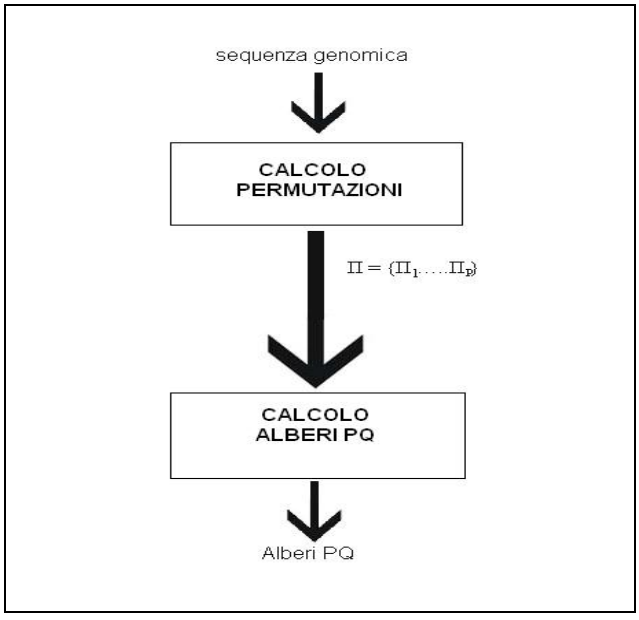

Figura 20

In figura 20, è mostrato un semplice schema dell'applicazione che, data in ingresso una sequenza genomica, calcola tutti gli alberi PQ. Essa si divide essenzialmente in due fasi:

- Nella prima fase, vengono calcolate (come discusso nel paragrafo precedente), tutte le permutazioni di lunghezza *n* presenti nella sequenza genomica. Ovvero, presa una stringa S_1 di lunghezza *n* all'interno della sequenza, vengono trovate tutte le altre stringhe S_i permutate presenti all'interno della sequenza stessa, venendosi cosi a crearsi l'insieme $\Pi=\{S_i|S_1=\pi_i(S_i)\ 1\leq i\leq t\}$, dove t sono le occorrenze di permutazioni trovate.

Figura 21

Nell'esempio in figura 21 con *n*=6, si ha S_1 = "*atcaat*"= $\pi_2(S_2$ = "*ttacaa*") = $\pi_3(S_3$ = "*caatat*"). Alla fine si avrà un insieme di insieme di permutazioni ovvero: $\Pi=\{\Pi_1, \Pi_2,\ldots.., \Pi_p\}$ dove ogni $\Pi_j=\{S_i \mid S_1=\pi_i(S_i)\ 1\leq i\leq t\}$.

- Nella seconda fase, dato l'insieme di insiemi di permutazioni $\Pi=\{\Pi_1,\Pi_2,..,\Pi_p\}$, vengono costruiti *p* alberi PQ, uno per ogni insieme di permutazioni mediante l'algoritmo per realizzare gli alberi PQ discusso nel capitolo precedente.

3.3 Prove effettuate

Verranno ora presentate le prove effettuate su delle sequenze genomiche, ortologhi delle sequenze umane e topesche, utilizzando gli alberi PQ. Sequenze, come detto in precedenza, in uso e studio all'ospedale Gaslini di Genova.

Tutti gli alberi PQ costruiti su i vari insiemi di permutazioni, sono stati classificati in base all'altezza, questo perché, come detto in precedenza, un albero PQ di bassa altezza, non dà alcuna informazione sulla natura e struttura delle varie stringhe che vanno a creare l'albero PQ, al contrario di un albero di maggiore altezza con una forma più complessa che individua una relazione tra le varie stringhe permutate che lo realizzano, fornendone la struttura combinatoria.

3.3.1 Costruzione alberi PQ su sequenza topo

In questa prova, è stata analizzata una sequenza genomica di un topo (mGluR1), dove, come spiegato in precedenza, prima sono stati calcolati gli insiemi di permutazioni (di dimensione n=200) presenti all'interno della sequenza, e su questi insiemi sono stati costruiti gli alberi PQ, su cui sono state calcolate le altezze per poter disporre di una classificazione di massima.

Gli insiemi di permutazioni trovati sono stati più di 25 mila (mediamente composti da una quindicina di stringhe permutate) ed i risultati su gli alberi PQ trovati sono rappresentati nella tabella 5.

ALTEZZA ALBERO	n° ALBERI
1	18018
2	13583
3	10897
4	3412
5	3513
6	514
7	281
8	8
9	5
10	3
11	5
12	1
13	1
14	0
15	1
16	0

Tabella 5

Come si può vedere, la maggior parte degli alberi costruiti sono di altezza 1, 2, 3 che strutturalmente hanno poca rilevanza per quanto riguarda la struttura delle permutazioni con cui sono state costruite. Interessante invece il fatto che man mano si sale di "altezza" diminuiscono i numeri di alberi trovati. In particolare analizziamo alcuni casi.

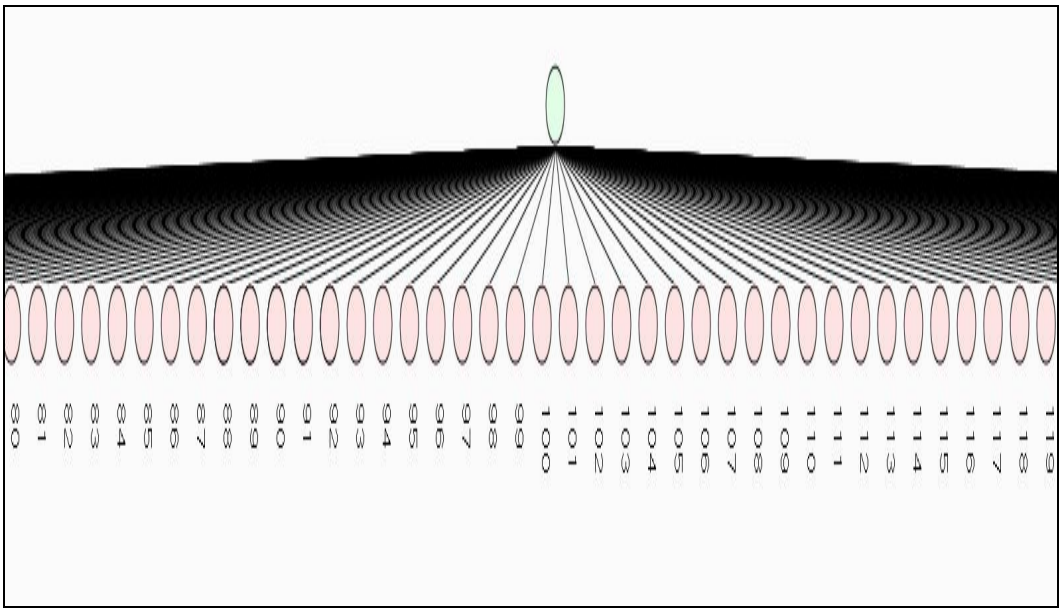

Figura 22: Albero altezza 1

La figura 22 rappresenta una parte di un albero di altezza 1. Esso è formato da un unico nodo (la radice) di tipo P e da 200 figli. Come detto in precedenza, esso non fornisce alcunché sulla struttura delle permutazioni che lo compongono. Quindi, permutazioni di questo tipo, possono tranquillamente essere scartate, in quanto, non contengono alcuna proprietà combinatoria di particolare interesse. Per vedere l'albero nella sua integrità, di seguito viene presentata la sua notazione scritta discussa nel capitolo precedente:

(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,193,194,195,196,197,198,199,200)

Ecco invece, l'insieme delle stringhe, trovate all'interno della sequenza genomica analizzata che compongono l'albero PQ di altezza 1:

- agaagacctaattccaactctcctcaaactattccacaaaatagaaacagaaggtattctacccaattcattctatgaagccacaattactctgatacctaaaccacacaaagatccaacaaagaaagagaacttcagaccaatttcccttatgaacatcgatgcaaaaatactcaataaaatcctcacaaaccgaatcc
- gaagacctaattccaactctcctcaaactattccacaaaatagaaacagaaggtattctacccaattcattctatgaagccacaattactctgatacctaaaccacacaaagatccaacaaagaaagagaacttcagaccaatttcccttatgaacatcgatgcaaaaatactcaataaaatcctcacaaaccgaatcca
- agacctaattccaactctcctcaaactattccacaaaatagaaacagaaggtattctacccaattcattctatgaagccacaattactctgatacctaaaccacacaaagatccaacaaagaaagagaacttcagaccaatttcccttatgaacatcgatgcaaaaatactcaataaaatcctcacaaaccgaatccaag
- gacctaattccaactctcctcaaactattccacaaaatagaaacagaaggtattctacccaattcattctatgaagccacaattactctgatacctaaaccacacaaagatccaacaaagaaagagaacttcagaccaatttcccttatgaacatcgatgcaaaaatactcaataaaatcctcacaaaccgaatccaaga

Si passa ora a vedere un albero di altezza maggiore, ovvero un albero con una struttura più interessante.

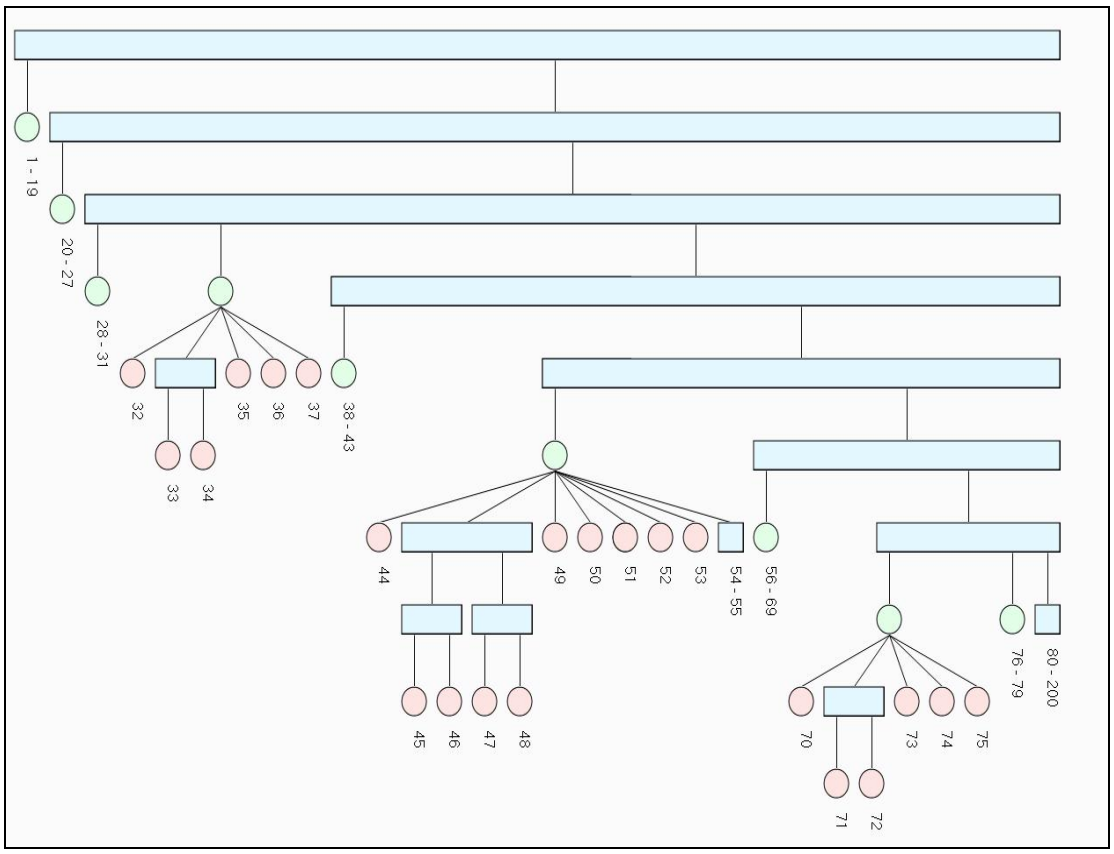

Figura 23: Albero di Altezza 15

La figura 23 rappresenta una parte di un albero di altezza 15. Come si può ben notare la struttura dell'albero è molto più complessa di un albero di altezza minore. Da ciò si può dedurre la particolare struttura interna e la relazione che le stringhe permutate del genoma, calcolate per la realizzazione dell'albero, possono avere. Questi insiemi di permutazione che generano questo tipo di alberi, sono molto interessanti, e vanno presi sicuramente in considerazione, perché mettono in evidenza alcune proprietà combinatorie della sequenza genomica analizzata, e si presentano in un numero limitato rispetto alla totalità delle ripetizioni di permutazioni, tale da poter mettere in evidenza alcune zone nel genoma studiato, con strutture rilevanti sotto il profilo permutativo.

Come fatto per l'albero di altezza 1, viene presentato l'albero PQ in forma testuale, e le relative stringhe che lo compongono. Si ricordi che la forma testuale dell'albero è una notazione ottenuta scrivendo l'albero PQ come una stringa *parentesizzata*, con differenti simboli per codificare un nodo P (usando la virgola come separatore "**,**") ed un nodo Q (usando come separatore il simbolo "**-**").

((1,2,3,4,5,6,7,8,9,10,((11-12)-(13-14)-(15-16)),17,18,19)-((20,21,22,23,24,25,(26-27))-((28,29,30,31)-(32,(33-34),35,36,37)-((38,39,40,41,42,43)-((44,((45-46)-(47-48)),49,50,51,52,53,(54-55))-((56,((57-58)-(59,60,61,62)),63,64,65,66,67,68,69)-((70,(71-72),73,74,75)-(76,77,78,79)-((80,81,82,83,84,85,86,87,88,89,90,((91-92)-(93-94)),95,96,97)-((98,(99-100),101,102,103,(104-105))-(106,107,108,109)-(110,((111-112)-(113-114)-(115-116)),117,118,119,((120-121)-(122-123)),124,125,126,127,128,(129-130),131,132,133,134,135,136,137,138,139,((140,141,142,143,144,145,146,147)-((148,(149-150),151,152,153)-(154,155,156,157,158,(159-160-161),162,163,164,(165,166,167,(168-169)),170,171,172,173,174,(175-176),177,178,179,(180,181,182,(183-184),185),186,187,(188,189,190,191,192),193,194,(195-196),197,198,199,200))))))))))))))

- atgtatttcgtgtgtgtgtatgtatacatgtatgtgtatgtatatgtgtgtatacatgtgtctgttcatatgtgtatgtatgtatttacttgtgtgtat gtgtacatgtatgtgtgtgtacacatacttgtgtatgagcatgtcatgatgtgtatgtgaaaatgaggacagcttacagctgtgggtgttctc ttcctact
- tgtatttcgtgtgtgtgtatgtatacatgtatgtgtatgtatatgtgtgtatacatgtgtctgttcatatgtgtatgtatgtatttacttgtgtgtatgt gtacatgtatgtgtgtgtacacatacttgtgtatgagcatgtcatgatgtgtatgtgaaaatgaggacagcttacagctgtgggtgttctctt cctacta
- gtatttcgtgtgtgtgtatgtatacatgtatgtgtatgtatatgtgtgtatacatgtgtctgttcatatgtgtatgtatgtatttacttgtgtgtatgt gtacatgtatgtgtgtgtacacatacttgtgtatgagcatgtcatgatgtgtatgtgaaaatgaggacagcttacagctgtgggtgttctctt cctactat
- tatttcgtgtgtgtgtatgtatacatgtatgtgtatgtatatgtgtgtatacatgtgtctgttcatatgtgtatgtatgtatttacttgtgtgtatgtgt acatgtatgtgtgtgtacacatacttgtgtatgagcatgtcatgatgtgtatgtgaaaatgaggacagcttacagctgtgggtgttctcttcc tactatg
- atttcgtgtgtgtgtatgtatacatgtatgtgtatgtatatgtgtgtatacatgtgtctgttcatatgtgtatgtatgtatttacttgtgtgtatgtgt acatgtatgtgtgtgtacacatacttgtgtatgagcatgtcatgatgtgtatgtgaaaatgaggacagcttacagctgtgggtgttctcttcc tactatgt

Si noti, dalle stringhe, la forte presenza di basi T e G, e la bassa presenza di C. Questo, come discusso nel primo capitolo, potrebbe far carpire qualche particolarità biologica che si presenta in quelle determinate zone della sequenza genomica, che si sono individuate mediante l'utilizzo degli alberi PQ.

3.3.2 Confronto tra sequenze genomiche

I risultati delle prove precedenti sono stati molto soddisfacenti sotto il profilo sia degli alberi costruiti sia delle poche zone con determinate caratteristiche di permutazione, trovate all'interno della sequenza genomica. Si è notato, però, che molte delle stringhe permutate che contribuivano alla costruzione degli alberi, presentavano una sovrapposizione del tipo mostrato in figura 16.

Fatto del tutto prevedibile dal momento che l'alfabeto che produce la sequenza è di piccola dimensione {A,C,G,T}, aggiunto all'enorme dimensione della sequenza. Da queste considerazioni si è pensato di sfruttare la potenzialità degli alberi PQ mettendo a confronto due diverse sequenze genomiche. L'idea di base è: date due sequenze genomiche DNA_1 e DNA_2, trovare tutte le permutazioni che oltre ad essere

presenti nella prima sequenza DNA_1, sono presenti contemporaneamente anche nella seconda sequenza DNA_2, come mostrato in figura 24.

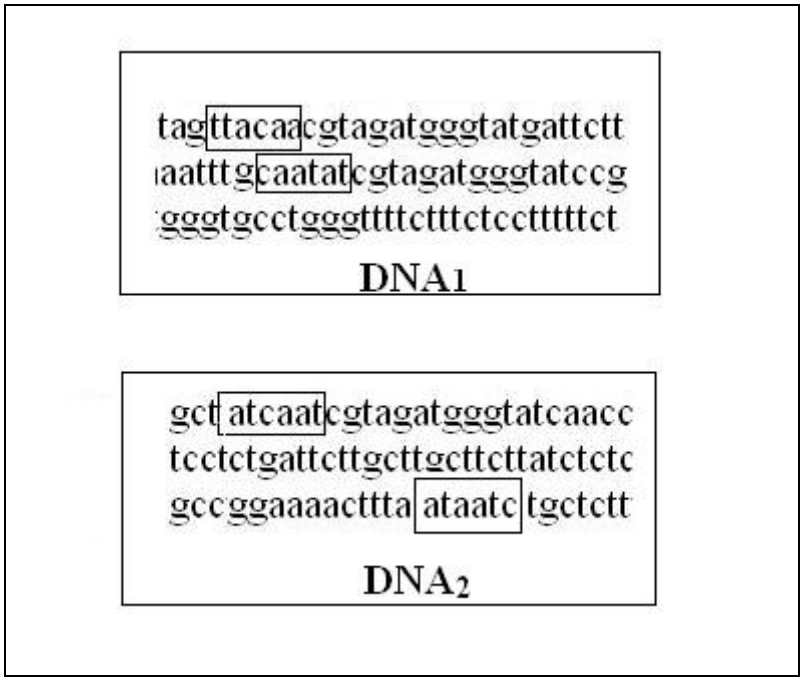

Figura 24

In altri termini, verrà calcolato l'insieme di insiemi di permutazioni $\Pi=\{\Pi_1,\Pi_2,\ldots..,\Pi_p\}$ dove ogni insieme:

$\Pi_i=\{ S_1,\ldots,S_t,S_{t+1},...,S_q \mid S_1 = \pi(S_i)\ 2\leq i\leq t \wedge (S_1, S_2,.S_t) \in DNA_1 \wedge (S_{t+1},S_2,..S_q) \in DNA_2\}$.

Come nella prima prova, sull'insieme di insiemi di permutazioni Π sono stati costruiti gli alberi PQ.

3.3.2.1 Confronto sequenze uomo-topo

In questo test sono state messe a confronto una sequenza genomica dell'uomo (grm1 human), ed una sequenza genomica del topo (mGluR1). Come nel primo test, su una singola sequenza, sono state calcolate le permutazioni (di dimensione n=200), generando un insieme di insiemi di permutazioni Π, e successivamente per ogni

insieme di permutazioni, contenuto in Π, è stato costruito un albero PQ. Come nel primo test, anche in questo caso è stata calcolata per ogni albero PQ creato, l'altezza, che verrà usata come fattore discriminate, per classificare tutti gli alberi realizzati.

ALTEZZA ALBERO	n° ALBERI
1	30870
2	10934
3	2161
4	1253
5	442
6	38
7	7
8	0
9	0
10	0

Tabella 6

Gli insiemi di permutazioni trovati, e quindi gli alberi costruiti, sono più di 45000 unità, ed i risultati del test sono presentati nella tabella 6.

Anche in questo test, come nel primo, si nota come man mano che l'altezza degli alberi aumenta, diminuiscono gli alberi trovati e che la maggior parte degli alberi, ha altezza 1 o 2. Questo permette già di fare una prima grossa scrematura degli alberi PQ realizzati (come analizzato in precedenza, alberi di altezza 1, 2 o 3 hanno poca rilevanza per quanto riguarda la struttura interna che hanno le permutazioni). In particolare si nota che gli alberi di altezza maggiore trovati (7) sono 7. Questo è un fatto di notevole rilevanza, perché mettere a confronto due sequenze genomiche e trovare, al loro interno, sottosequenze che sono permutate tra loro, ed allo stesso tempo, che godono di particolari proprietà combinatorie, dimostrato dalla teoria degli alberi PQ, tali da distinguerle dal resto delle sottosequenze, mettendo in luce

possibili zone di particolare interesse biologico, risulta essere particolarmente importante per l'analisi e lo studio delle sequenze genomiche.

Viene mostrato adesso un esempio in figura 25.

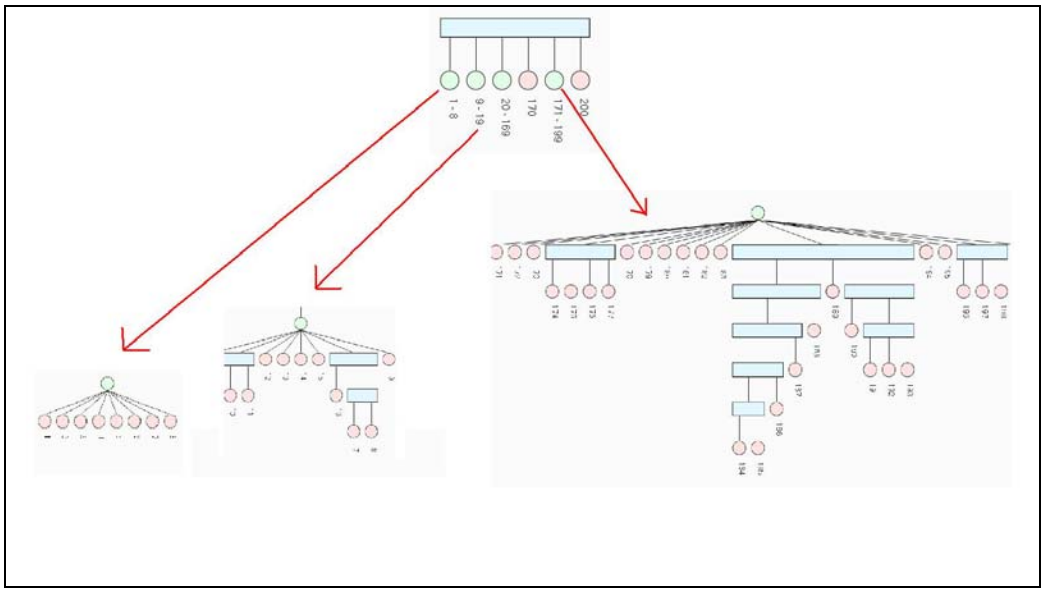

Figura 25

Ecco anche la relativa forma testuale dell'albero PQ di altezza 7 e le stringhe permutate che lo compongono:

((1,2,3,4,5,6,7,8)-(9,(10-11),12,13,14,15,(16-(17-18)),19)-(20,21,22,23,24,25,26,(27-28),29,(30-31),32,33,34,35,(36-37),38,39,40,41,42,43,44,45,46,47,48,49,(50-51),(52-53),54,55,56,57,58,(59-60),61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,(79-(80-81)),82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,(109-110),111,112,113,114,115,116,117,118,(119-120),121,122,(123-124),125,(126-127),128,(129-130),131,132,133,134,135,(136-137-138),139,(140-141),142,143,144,145,146,147,(148-149),(150-151),152,153,154,155,156,157,158,(159-160),161,162,(163-164),165,166,167,168,169)-170-(171,172,173,(174-175-176-177),178,179,180,181,182,183,(((((184-185)-186)-187)-188)-189-(190-(191-192-193))),194,195,(196-197-198),199)-200)

- acattcaacttcatgcttgttagtctttctttgtcttactgattttaaattggcatgaatccctaatgctcacctgtataatataccctctcaattcctatatacaattctgaccccaagatttttttctgcatttgttgcagccttaattccactttctttcttcttcttttttaaatattggatagtttctttat
- ccaaatatttgcctgattcactccctattcttcttcaattattcactcaatgctcctctttttagtgaggtctactctaaccaatcgacatgatactgtctcaacatacccactttttctttcctgttctcttgtatttttttctgtaacactgatttgattgtaattattacataattttcttagttgtgttaatttttt

Delle due stringhe che compongono l'albero PQ di altezza 7, la prima fa parte della sequenza del topo, mentre la seconda appartiene alla sequenza umana. Oltre all'altezza dell'albero, è stata analizzata anche la frequenza delle basi A, C, G, T presenti nelle stringhe permutate. Questo può aiutare, come accennato nel primo capitolo, a capire particolari funzionalità di quelle determinate zone (le stringhe permutate trovate) individuate nelle sequenze genomiche grazie all'aiuto degli alberi PQ. Nell'esempio mostrato, si ha una maggiore presenza di T a discapito di una poca presenza di G.

3.3.2.2 Altre prove effettuate

In questo paragrafo, verranno brevemente presentati i risultati di altre due prove svolte sulla falsa riga dei test descritti nel paragrafo precedente. Si differenziano dal precedente test, solo per l'utilizzo di diverse sequenze di DNA. La prima prova, come la precedente, è stata effettuata confrontando una sequenza di DNA umano con una sequenza di DNA di topo, ma utilizzando due sequenze diverse dalla prova precedente. I risultati ottenuti sono riportati in tabella 7.

ALTEZZA ALBERO	n° ALBERI
1	17865
2	6054
3	1189
4	629
5	214
6	20
7	6
8	0
9	0
10	0

Tabella 7

Anche in questo caso, sono stati confermati i risultati del test precedente, fatto con diverse sequenze. Molti alberi di piccola altezza, privi di alcun significato, a differenza dei pochi alberi di altezza maggiore (ad esempio 6 alberi di altezza sette), che oltre a fornire l'esistenza di una particolare relazione che vi è tra le sottosequenze trovate, che formano gli alberi di altezza maggiore, ci aiutano ad identificare particolari zone all'interno delle due sequenze con determinate proprietà permutative come descritto nel test precedente.

L'altro test effettuato si riferisce ad un confronto tra due diverse sequenze genomiche di topo. Nella tabella 8 sono presentati i risultati:

ALTEZZA ALBERO	n° ALBERI
1	54506
2	16160
3	3405
4	1814
5	633
6	51
7	9
8	0
9	0
10	0

Tabella 8

I risultati dei test precedenti vengono rispecchiati anche in quest'ultimo test. Si noti come in tutti e tre i test il limite massimo per l'altezza degli alberi costruiti è sempre 7, mettendo in evidenza, l'esistenza di determinate zone all'interno delle sequenze del DNA con particolari proprietà permutative.

3.4 Validazione dei risultati

Si potrebbe pensare che i risultati ottenuti dalle tre prove precedenti, siano dovuti al caso, e che gli alberi di una certa altezza (e la loro distribuzione) nascono casualmente. Quindi si è deciso, per dare una validazione scientifica, di effettuare due ulteriori test per capire se gli alberi di altezza 7, ad esempio, siano significativi e nascono da un determinato motivo biologico.

I due nuovi test effettuati sono stati realizzati nel seguente modo:

1. Vengono create le sequenze, DNA_1', ottenuta permutando casualmente tutte le basi di una sequenza genomica utilizzata nei precedenti test, DNA_1, e DNA_2' ottenuta sempre permutando a caso tutti i caratteri di DNA_2 (un'altra sequenza genomica, utilizzata sempre nei precedenti test). Si effettuerà il test confrontando, similmente al test effettuato nel precedente paragrafo, le sequenze genomiche DNA_1' e DNA_2'.
2. DNA_1' è ottenuta come descritto nel punto precedente, e viene confrontata con la sequenza DNA_2.

Le prove sono state effettuate utilizzando le sequenze genomiche utilizzate nei primi test. Eccone i risultati:

ALTEZZA ALBERO	n° ALBERI
1	326
2	187
3	47
4	29
5	12
6	0
7	0
8	0
9	0
10	0

Tabella 9: Confronto DNA_1' DNA_2'

ALTEZZA ALBERO	n° ALBERI
1	299
2	26
3	3
4	2
5	1
6	0
7	0
8	0
9	0
10	0

Tabella 10: Confronto DNA_1' DNA_2

Entrambi i test forniscono dei risultati davvero sorprendenti che confermano quanto fatto nelle precedenti prove. Innanzitutto si noti come il numero delle permutazioni sia calato drasticamente, (gli insiemi di permutazione trovati nelle prime prove si aggiravano mediamente sulle 30 mila, in queste ultime due prove si hanno rispettivamente 602 e 332 insiemi di permutazioni ovvero alberi costruiti), e questo fa pensare innanzitutto che le permutazioni che sono state trovate nelle sequenze genomiche hanno una funzione ben precisa, e non sono, dunque, dovute al caso. Lo stesso discorso, poi, può essere fatto sugli alberi costruiti. In entrambi i test di validazione non sono stati trovati alberi con altezza maggiore o uguale a 6 nelle

sequenze dettate del tutto dal caso, alberi di particolare struttura (ad esempio altezza 7) che vanno ad identificare particolari zone all'interno delle sequenze di DNA.

Questo fa capire l'importanza dei test precedentemente effettuati e soprattutto dei risultati ottenuti, che hanno permesso di mettere in evidenza, alcune zone all'interno delle sequenze genomiche, che si differenziano, per le loro caratteristiche permutative, dal resto delle sottosequenze presenti all'interno del genoma. In più, l'uso degli alberi PQ, permette di mettere a punto una nuova metodologia, capace di mettere in risalto pattern ripetuti all'interno di sequenze di DNA, tali da consentire di estrarre informazioni rilevanti dal punto di vista biologico.

CAPITOLO 4

Conclusioni

Negli ultimi anni, l'analisi delle sequenze di DNA, ha avuto un ruolo sempre più importante, divenendo un tema di grande attualità. Il numero di sequenze biologiche a disposizione è sempre maggiore, e sommando le enormi dimensioni dei dati da analizzare, si ha il bisogno di avere strumenti in grado di identificare regioni del genoma ricco di particolari funzionalità. Un metodo *pattern discovery*, si pone l'obiettivo di individuare regioni simili all'interno di due o più sequenze per scoprire origini e funzioni evolutive comuni.

Questa tesi si è posta l'obiettivo di fornire un nuovo metodo per la scoperta, all'interno delle sequenze genomiche, di zone con particolari funzioni biologiche. Si è partiti con un classico studio delle permutazioni contenute all'interno delle sequenze analizzate, ma non bastando, serviva un metodo per poter filtrare le permutazioni più interessanti, a discapito di quelle meno interessanti. Questo è stato effettuato mediante l'introduzione e l'utilizzo della struttura dati degli alberi PQ. Grazie al loro utilizzo, si è riusciti ad individuare e mettere in evidenza alcune zone delle sequenze genomiche analizzate, che presentavano particolari proprietà permutative.

Ovviamente bisognerà effettuare uno studio biologico più approfondito per vedere se le sottosequenze trovate, oltre ad avere proprietà permutative ed essere identificate

facilmente mediante l'utilizzo degli alberi PQ, hanno determinate proprietà biologiche.

Riferimenti Bibliografici

[1] R. J Robbins. *Challenges in the Human genoma project*. IEEE Engineering in Medicine and Biology, 11, 25-34, 1992.

[2] B. Alberts. *Molecular Biology of the Cell. Garland Science*, 5th edition, 2007.

[3] B. Lewin. *Genes IX.* Prentice Hall, 2007.

[4] A. Rich. *DNA comes in many forms.* Gene 135: 99-109, 1993.

[5] A. La Terza, V. Passini, S. Barchetta. *Adaptive evolution of the heat-shock response in the Antarctic psychrophilic ciliate, Euplotes focardii: hints from a comparative determination of the hsp70 gene structure*. ANTARCTIC SCIENCE, 19: 239-244. 2004

[6] M. G. Kidwell. *Transposable elements. in: The Evolution of the Genome*. Elsevier, 165-221. 2005.

[7] G.B. Singh. *Discovering Matrix Attachment Regions (MARs) in genomic databases*. ACM SIGKDD Explorations Newsletter, 1, 39-45. 2000.

[8] K. Liolios, K. Mavromatis, N. Tavernarakis, N.C. Kyrpides. *The Genomes On Line Database (GOLD) in 2007: status of genomic and metagenomic projects and their associated metadata.* N.A.R., 36, Database issue: D475-D479. 2008.

[9] Yuan Gao and Kalai Mathee. *Motif Detection in Protein Sequences*. SPIRE/CRIWG 1999.

[10] Maxime Crochemore, Costas S. Iliopoulos, Manal Mohamed and Marie

France Sagot. *Longest repeats with a block of k don't care*. Theorical Computer Science 362 248-254. 2006.

[11] I. Rigoutsos and A. Floratos. *Combinatorial pattern discovery in biological sequences : the TEIRESIAS algorithm*. BIOINFORMATICS vol.14 no. 1, pages 55-67, 1998.

[12] Martin Ester and Xian Zhang. *A Top-Down Method for Mining Most Specific Frequent Patterns in Biological Sequence Data*. Fourth SIAM International Conference on Data Mining, Lake Buena Vista (Florida, USA) 2002.

[13] Laxmi Parida, Isidore Rigoutsos, Aris Floratos, Dan Platt, Yuan Gao. *Pattern Discovery on Character Sets and Real-valued Data: Linear Bound on Irredundant Motifs and an Efficient Polynomial Time Algorithm*. SODA 2000: San Francisco (California, USA). Pages 297-308, 2000.

[14] I. Jonassen, J. F. Collins and D. G. Higgins. *Finding flexible patterns in unaligned protein sequences.* Protein Science number 4, pages 1587-1595. 1995.

[15] T. Uno and M. Yagiura. *Fast algorithms to enumerate all common intervals of two permutations.* In Algorithmica, 26(2):290-309, 2000.

[16] K. Booth e G. Leuker. *Testing for the consecutive ones property, intervals graphs, and graph planarity using pq-tree algorithms*. In Journal of Computer and System Sciences, 13:335-379, 1976.

[17] S. Heber and J. Stoye. *Finding all common intervals of k permutations.* In Proceedings of the 12th Annual Symposium on Combinatorial Pattern Matching (CPM), 207–218, 2001.

[18] G.M. Landau, L. Parida, O. Weimann. *Using PQ trees for comparaive genomics.* In proceedings of 16th annual Symposium on Combinatorial Pattern Matching (CPM). 2005.

[20] R. Eres, G.M Landau, L. Parida. *Permutation Pattern Discovery in biosequences.* In Journal of computational Biology, 1050-1056. 2004.

[21] A. Amir, A. Apostolico, G.M. Landdau, G.Satta. Efficient *Text Fingerprinting via Parikh Mapping*. Journal of Discrete Algorithms Volume 1, Number 5, 409-421(13). 2003.

[22] J.D. Tissdall. *Mastering Perl for Bioinformatics*. 2003.

Ringraziamenti

Volevo ringraziare innanzitutto il Prof. Roberto Grossi e la Dott. Nadia Pisanti, per avermi dato la possibilità di svolgere questo lavoro, per la disponibilità e l'aiuto datomi. Un ringraziamento anche al Dott. Oren Weimann per la disponibilità e la collaborazione.

Un grazie particolare a mio padre e mio fratello per il prezioso aiuto e per avermi saputo incoraggiare nei momenti difficili.

Un ringraziamento particolare a Mirella per il sostegno che mi ha dato in ogni momento e per aver sempre creduto in me e nelle mie capacità. Senza il suo supporto non avrei mai raggiunto questo traguardo.

Ringrazio anche tutti gli amici che in questi anni hanno rallegrato e reso più leggeri, i lunghi periodi di studio.

Grazie a tutti coloro che oggi condividono con me questa grandissima gioia.