GHEORGHE-IOAN MIHALAŞ
ANCA TUDOR
SORIN PARALESCU

BIOINFORMATICA

Colecția

ŞTIINȚELE EXACTE ÎN CERCETAREA MEDICALĂ

GHEORGHE-IOAN MIHALAŞ
ANCA TUDOR
SORIN PARALESCU

BIOINFORMATICA

2011

Editura VICTOR BABEŞ
Piața Eftimie Murgu 2, cam. 316, 300041 Timişoara
Tel./ Fax 0256 495 210
e-mail: *evb@umft.ro, calaevb@umft.ro*
www.evb.umft.ro

Director general: Prof. univ. dr. Ştefan-Iosif Drăgulescu
Consilier editorial: Claudia Christian
Coperta: Caius Fericean

Colecția: ŞTIINȚELE EXACTE ÎN CERCETAREA MEDICALĂ
Coordonator colecție: Prof. univ. dr. Gheorghe-Ioan Mihalaş
Referent ştiințific: Prof. univ. dr. Diana Lungeanu

Descrierea CIP a Bibliotecii Naționale a României
MIHALAŞ, GHEORGHE-IOAN
Bioinformatică / Gheorghe-Ioan Mihalaş, Anca Tudor, Sorin Paralescu. - Timişoara: Editura Victor Babeş, 2011
ISBN 978-606-8054-33-9

I. Tudor, Anca
II. Paralescu, Sorin

57:004

Tipărit la Tipografia Eurostampa
B-dul Revoluția din 1989 nr. 26, Timişoara
Tel. 0256- 204 816, edituraeurostampa@gmail.com

Prefață

Bioinformatica a apărut natural, ca disciplină înrudită cu informatica medicală (ulterior integrată în ea), dedicată stocării şi prelucrării datelor din biologie. În ultimele decenii, aceste date au început să conțină tot mai multe detalii structurale, în special secvențe moleculare. Informațiile noi, care soseau cu o cadență accelerată impuneau cu stringență nu numai noi abordări în organizarea şi sistematizarea datelor şi algoritmi performanți de prelucrare, ci şi necesitatea pregătirii unor specialişti în bioinformatică. Aplicațiile din ce în ce mai numeroase în domeniul medical şi farmaceutic au ridicat şi mai mult presiunea, astfel încât în ultimul deceniu al secolului trecut au apărut primele calificări, la nivel masteral, în bioinformatică. Trebuie însă remarcat caracterul interdisciplinar deosebit al bioinformaticii, acoperind o paletă mult mai largă decât alte discipline de graniță. Tendința de îngustare a domeniului de specializare, manifestată pregnant în învățământul superior în ultima parte a secolului XX a redus sensibil aria de recrutare a specialiştilor în toate ramurile multidisciplinare, iar pentru bioinformatică această reducere a devenit drastică. Comunitatea specialiştilor bioinformaticieni, deşi într-o rapidă creştere, este încă destul de redusă – probabil aşa se explică de ce această specialitate este cea mai bine remunerată dintre toate specialitățile informaticii medicale.

Ce este deosebit la această interdisciplinaritate? Aşa cum am afirmat mai sus - paleta largă de intersecție a unor discipline, uneori destul de îndepărtate între ele. Într-adevăr, centrul atenției se îndreaptă spre principalele molecule de interes informațional: acizi nucleici şi proteine. Deci este necesară o bază solidă de biologie moleculară, care este ea însăşi un domeniu interdisciplinar, necesitând o serie de cunoştințe de biochimie şi de biologie celulară. Pentru aplicațiile medicale şi interpretarea hărților cromozomiale sunt necesare o serie de cunoştințe de genetică. Să ne îndreptăm privirea şi în cealaltă parte, privind dezvoltarea metodelor de prelucrare. Sunt necesare cunoştințele solide de teoria probabilităților şi statistică, topologie sau analiza seriilor. Determinarea structurilor tridimensionale şi a aplicațiilor farmaceutice de "drug design" sunt aplicații ale chimiei cuantice. O serie de noțiuni de fizică şi biofizică sunt de asemenea necesare, atât pentru înțelegerea proprietăților structurilor analizate cât şi a metodelor utilizate pentru obținerea şi interpretarea datelor experimentale. Să nu uităm nici capitolul de biologia sistemelor, în care sunt urmărite procesele de semnalizare celulară, studiate de fiziologia celulară şi prelucrate prin simulare atât prin ecuații diferențiale cât şi simulare stochastică. Pentru viziunea integrativă sunt necesare noțiuni din teoria sistemelor, cibernetică şi teoria evoluționismului. Desigur, să adăugăm în final şi câteva discipline fundamentale din ştiința calculatoarelor – baze de date şi limbaje de programare.

Nu este deci de mirare că, la înscrierea la cursurile de bioinformatică, în universitățile ce au introdus aceste cursuri, se presupune că studenții au parcurs anterior cursuri de biofizică, biochimie, biologie moleculară şi celulară sau genetică şi au baza matematică adecvată.
Având în vedere complexitatea pregătirii în atâtea domenii adiacente, prezentul manual a fost alcătuit astfel încât şi absolvenții unui profil tehnic să-şi găsească aici şi noțiunile fundamentale din aceste discipline. În această viziune am inclus, după primul capitol privitor la noțiuni generale de informatică medicală, un set de patru capitole dedicate prezentării sumare a noțiunilor utile pentru abordările din partea a doua.
În ultima parte sunt prezentate lucrările practice de laborator, esențiale pentru a simți adevărata dimensiune a imenselor baze de date moleculare şi complexitatea metodelor de prelucrare.
Deşi această carte – prima de acest fel din țară – a fost elaborată mai ales pentru studenții la nivel masteral în profilul "Sisteme informatice în îngrijirea sănătății" (SIIS) de la Universitatea Politehnica, respectiv profilul "Biologie moleculară şi biotehnologii" (BMB), de la Universitatea de Medicină şi Farmacie, ea se adresează de fapt tuturor celor interesați de acest domeniu fascinant, care va oferi încă numeroase surprize. Lumea vie are încă multe secrete nedescifrate!
Închei prefața citând un student (într-un sondaj despre curs): "Este mai interesant ca la 'Discovery'!"

Prof. dr. G. I. Mihalaş
membru al Academiei de Ştiințe Medicale

CUPRINS

Partea I *(G.I.Mihalaş)*

Partea a II-a *(A. Tudor, S. Paralescu)*

Partea I

1. Introducere în Informatica Medicală

1.1. Obiectul Informaticii Medicale

1.1.1. Scurt istoric

A. Informatica medicală este o disciplină tânără, termenul apărând în cursul anilor ’60. În accepțiunea inițială informatica medicală cuprindea *programele de calculator* cu aplicabilitate în domeniul medical. Progresul tehnic rapid a arătat însă că, pentru aceleaşi aplicații, atât programele cât şi suportul fizic se schimbau; ceea ce rămânea la fel era modul în care era prelucrată *informația.*

B. Astfel, în accepțiunea actuală, centrul definiției s-a mutat de la „calculator” la informație. Coiera [1997] chiar atrage atenția în acest sens: „Informatica medicală se ocupă de calculatoare tot atât de mult cât se ocupă cardiologia de stetoscoape”.

1.1.2. Obiectul informaticii medicale

Accepțiunea clasică: totalitatea programelor de ***calculator*** cu aplicații în domeniul biomedical şi sănătate.

Definiția actuală: disciplina care studiază întregul flux al ***informației medicale:*** generare, achiziție, stocare, transmitere, prelucrare şi utilizare.

1.2. Teoria informației

1.2.1. Noțiunea de informație

Pentru a ne ocupa de *informația medicală*, să încercăm mai întâi să privim conceptul de *informație* la modul general.

A. Termenul de informație este folosit în mod curent în viața de zi cu zi, fiind cel mai adesea asociat cu aducerea unui element de noutate. Fiind un concept cu grad înalt de generalitate (categorie filosofică), informația nu poate fi definită în manieră clasică, pornind de la genul proxim şi precizând diferențele specifice, ci prin proprietatea sa esențială – cea de a înlătura o nedeterminare.

B. Noțiunea de informație

Informația este un concept cu grad înalt de generalitate caracterizat prin proprietatea de a înlătura o nedeterminare (incertitudine).

1.2.2. Proprietățile informației

Informația nu este materie; totuşi ea nu poate exista înafara materiei. Norbert Wiener spunea „Creierul nu secretă informație precum ficatul fiere”.

Informația nu este energie; totuşi ea nu se poate transmite fără un suport energetic.

Nu este o relație directă între cantitatea de energie ce însoțeşte transmiterea unei informații şi cantitatea de informație transmisă. De exemplu, energia unui trăsnet în

timpul unei furtuni este imensă, însă informaţia transmisă este neglijabilă; în schimb un foşnet într-o pădure, purtat de o energie infimă poate reprezenta o informaţie vitală – punând pe fugă un animal! Să semnalăm totuşi că nu există relaţie nici între cantitatea de informaţie şi efectele sale. De ex. Legenda lui Tezeu din metodologia greacă: Tezeu promisese tatălui său Aegeus, că dacă va învinge în luptă minotaurul va înlocui pânza neagră a corabiei cu pânză albă, dar a uitat şi tatăl său s-a aruncat de pe stânci. Informaţia primită a fost doar 1 bit.

1.2.3. Triada abordărilor complete

A. Introducerea aspectelor informaţionale în studiul materiei vii completează imaginea noastră privind complexitatea sistemelor biologice, actualmente considerându-se că o abordare completă trebuie să acopere atât aspectele materiale şi energetice cât şi cele informaţionale.

B. Triada abordărilor complete:
- aspectul material – structura
- aspectul energetic – suportul funcţional
- aspectul informaţional – mecanismul funcţional.

C. Valoarea utilă a informaţiei depinde de receptor.

Sensul noţiunii de informaţie, aşa cum a fost prezentat mai sus etc. legat de altă noţiune – nedeterminarea (sau incertitudinea) – vag definită la rândul său. Acelaşi mesaj poate să aibă valori informaţionale diferite pentru diferiţi receptori: pentru o persoană care deja ştia conţinutul său cantitatea de informaţie primită este zero, însă pentru receptorii care nu-i ştiau conţinutul va putea fi evaluată cantitatea de informaţie primită.

1.2.4. Cantitatea de informaţie

Parcurgerea acestui subiect necesită cunoştinţe fundamentale de teoria probabilităţilor

A. Pornind de la proprietatea fundamentală a informaţiei, cea de a înlătura o nedeterminare, Shannon a considerat că informaţia primită este invers proporţională cu probabilitatea de apariţie a evenimentului: dacă se va întâmpla un eveniment cu probabilitate mare, informaţia primită este mică; în schimb primim o informaţie „mai mare" dacă apare un eveniment mai rar. (Ziariştii exploatează intens această relaţie!)

B. Relaţia propusă de Shannon pentru calculul cantităţii de informaţie care este primită când se petrece un eveniment cu probabilitatea p_i cuprinde logaritmul în baza 2 din inversul probabilităţii p_i.

- Cantitatea de informaţie eliberată de un eveniment a cărui probabilitate este p_i

$$I_i = -\log_2 p_i \qquad (1.2.4.a)$$

C. Pe baza acestei relaţii stabileşte unitatea de măsură pentru cantitatea de informaţie, numită *bit* (de la BInary digiT).

Definiţie: un bit este cantitatea de informaţie primită când se înlătură o nedeterminare de ½.

D. În mod uzual informaţia se transmite printr-o succesiune de evenimente, numită adesea *mesaj*, iar un eveniment într-un mesaj se mai numeşte *simbol*.

În cazul unui mesaj format din N evenimente, fiecare eveniment *i* apare de n_i ori, aducând de fiecare dată informaţia I_i, deci mesajul aduce informaţia.

$$I = n_1 I_1 + \eta_2 I_2 + \ldots + n_k I_k = \sum_{i=1}^{k} n_i I_i \qquad (1.2.4.b)$$

E. Valoarea medie a informației corespunzătoare unui eveniment într-un şir de N evenimente se mai numeşte „entropie informațională", H, şi se calculează astfel:

$$H = \frac{n_1 I_1 + n_2 I_2 + \ldots + n_k I_k}{N} = \frac{n_1}{N} I_1 + \frac{n_2}{N} I_2 + \ldots + \frac{n_k}{N} I_k$$

la limită (i.e. atunci când $N \to \infty$) relația devine:

$$H = p_1 I_1 + \ldots + p_k I_k = \sum_{i=1}^{k} p_i I_i$$

Înlocuind I_i conform relației (1.2.4.a), obținem formula (1.2.4.c), formulă fundamentală în teoria informației, numită şi formula lui Shannon pentru entropia informațională.

- Entropia informațională este cantitatea medie de informație per eveniment (simbol) într-un mesaj:

$$H = -\sum_{i=1}^{k} p_i \log_2 p_i \qquad (1.2.4.c)$$

1.2.5. Relația între entropia informațională şi entropia termodinamică

A. Termenul de „entropie" a fost introdus în termodinamică pentru enunțarea principiului al II-lea al termodinamicii: „În procesele termodinamice entropia nu poate să scadă: ea rămâne constantă în cadrul proceselor reversibile şi creşte în cazul proceselor ireversibile".

B. Relația între entropia termodinamică şi cea informațională poate fi înțeleasă pornind de la experimentul „ideal" propus de Maxwell pentru explicarea variației entropiei în cazul proceselor ireversibile, prezentat în cadrul cursului de biofizică.

C. Se vede deci că sistemul poate evolua în sens contrar celui dictat de al II-lea principiu al termodinamicii în cazul în care primeşte o informație. Acesta este mecanismul prin care sistemele vii evoluează spre stări tot mai organizate şi deosebite de mediul înconjurător.

1.2.6. Redondanță

A. Entropia informațională are valoare maximă când evenimentele din mesaj sunt echiprobabile: $p_i = 1 / k$;

atunci $H_{\max} = k\left(\frac{1}{k}\log\frac{1}{k}\right)$ de unde se obține relația (1.2.6.a)

$$H_{max} = log_2 k \qquad (1.2.6.a)$$

B. Un exemplu ar fi cazul unui mesaj encriptat, în care probabilitatea apariției unui simbol este (cel puțin aparent) independentă de simbolurile anterioare. În mesajele

reale probabilitatea unui simbol depinde de simbolurile anterioare; putem, în funcție de context, să „ghicim" ce urmează, putem folosi prescurtări, putem observa greşeli cum ar fi omisiunea unei litere etc. Deci informația nu este distribuită uniform în mesaj sau chiar în interiorul cuvintelor, cantitatea de informație transportată în realitate fiind inferioară celei maxime ce ar putea fi transmise folosind aceeaşi lungime a textului. Această diferență, între cantitatea maximă ce poate fi conținută în mesaj şi cea reală se numeşte redondanță şi reprezintă o parte din mesaj care… nu conține informație!

C. Relația de definiție a redondanței absolute este:

$$R = H_{max} - H_{real} \qquad (1.2.6.b)$$

Raportând redondanța absolută la H_{max} se defineşte redondanța relativă.

$$R_r = R / H_{max} \qquad (1.2.6.c)$$

D. Rolul redondanței

Aparent redondanța ar reprezenta o încărcătură inutilă în mesaj. Totuşi, prezența ei diminuează rolul negativ al perturbațiilor ce apar în cursul transmiterii informației, putând deseori reconstitui mesajul inițial chiar dacă unele simboluri au fost perturbate.

1.3. Transmiterea informației

1.3.1. Sisteme de comunicație

A. Am precizat anterior că valoarea utilă a informației depinde de receptor, deci noțiunea de informație are sens doar dacă se transmite; altfel, rămâne în faza de „informație potențială".

B. Transmiterea informației presupune o *sursă* a informației (emițător E) şi un destinatar (receptor R). Spațiul dintre S şi R reprezintă *canalul de comunicație* (C). Pe canalul de comunicație pot să apară diverse *zgomote* care perturbă sistemul de comunicație afectând calitatea transmisiei (figura 1.3.1.a).

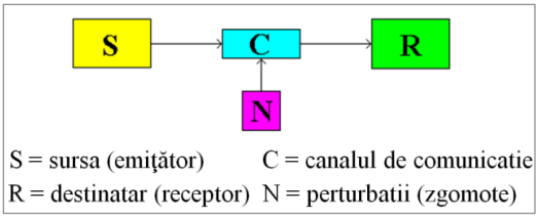


Fig. 1.3.1.a. Sistem de comunicație simplu

C. Să introducem doi termeni importanți în cadrul sistemelor de comunicație:

- *mesaj* – un termen pe care îl folosim când ne referim la conținutul informațional al transmisiei
- *semnal* – suportul fizic care transportă mesajul (sunet, curent electric etc.).

D. Pentru diminuarea efectelor perturbațiilor sau pentru a asigura transmiterea mesajului la distanțe foarte mari, se introduc pe canalul de transmisie nişte dispozitive numite *traductori*. Un traductor schimbă suportul fizic al unui semnal (figura 1.3.1.b). De exemplu, în cazul unei convorbiri telefonice, microfonul este traductorul localizat lângă emițător, transformând sunetele (variații ale presiunii aerului) în variații ale unui curent electric. Canalul de comunicație este reprezentat de firele telefonice. La

destinatar un alt traductor, casca telefonică, transformă variaţiile curentului electric în vibraţii ale unei membrane elastice generând astfel sunete.

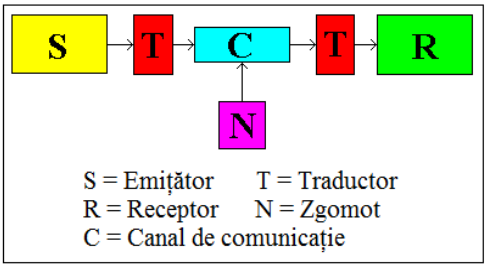


Fig. 1.3.1.b. Schema generală a unui sistem de comunicaţie

E. Există şi alte dispozitive ce pot fi utilizate în sisteme de comunicaţie, de ex. *modem-ul*. Denumirea modem provine de la modulator/demolator. Modemul este un dispozitiv care asigură modularea semnalului, adică suprapunerea semnalului real peste un semnal purtător (undă purtătoare) care are caracteristici încât se diminuează efectul perturbaţiilor (de ex. perturbaţiile uzuale, de joasă frecvenţă, sunt eliminate dacă unda purtătoare are frecvenţă înaltă).

F. O altă transformare pe care o putem aplica semnalului pentru transmisie este *codificarea*. Mesajul este compus uzual dintr-o succesiune de simboluri. Totalitatea simbolurilor utilizate pentru a compune un mesaj se numeşte *alfabet.* Simbolurile alfabetului se mai numesc „*litere*”, iar cu literele putem construi *cuvinte.* Totalitatea cuvintelor cu sens reprezintă un *dicţionar*, iar precizarea sensului cuvintelor se numeşte *semantică*. Cu ajutorul cuvintelor se pot construi propoziţii; regulile de construcţie a propoziţiilor se numeşte *sintaxă.* Un dicţionar împreună cu semantica şi o sinteză reprezintă un *limbaj*. Noi folosim uzual pentru comunicaţie *limbaje naturale*, dar există posibilitatea utilizării unor *limbaje formale* sau *artificiale.* Diferitele componente ale sistemului de comunicaţie pot folosi diferite alfabete sau dicţionare. Transpunerea unui mesaj dintr-o formă ce utilizează un alfabet într-o formă în alt alfabet, cu anumite reguli de corespondentă se numeşte *codificare*. Operaţiunea inversă se numeşte *decodificare.* Transpunerea unui mesaj dintr-un limbaj în altul se numeşte *traducere*.

G. Să mai menţionăm legat de sistemele de comunicaţie că există o capacitate limitată de transmisie a informaţiei pe canalul de comunicaţie, numită *viteză de transmisie*, măsurată în bit/secundă.

1.3.2. Transmiterea informaţiei în materia vie

A. *Codul genetic*. Informaţia privind structura proteinelor ce pot fi sintetizate este stocată în molecula de ADN din nucleu. Acizii nucleici conţin 4 baze azotate: adenina A, timina T, citozina C şi guanina G (în cazul ARN în loc de timină apare uracilul U). Proteinele sunt formate din 20 de aminoacizi esenţiali. O succesiune de 3 baze azotate din ADN se numeşte *codon* şi poartă informaţia pentru codificarea unui aminoacid într-o secvenţă proteică. Totalitatea corespondenţilor între codoni şi aminoacizii corespunzători poartă denumirea de *cod genetic*. Porţiunea dintr-un lanţ ADN care poartă informaţia pentru sinteza unei proteine se numeşte *genă*, iar ansamblul tuturor genelor unei specii se numeşte genom. Genomul uman conţine circa 30.000 gene. Prezentul curs este dedicat acestui tip de informaţie.

B. *Codificarea informației în sistemul nervos.* Pe axoni informația este transmisă printr-o succesiune de impulsuri nervoase; fiecare impuls nervos este un potențial de acțiune care are întotdeauna aceeași amplitudine. Unui stimul mai intens îi corespunde o rată mai ridicată de formare a potențialelor de acțiune; spunem că pe axon informația privind intensitatea stimulului este *codificată în frecvență*.

La nivelul sinapselor are loc o descărcare a veziculelor cu mediator chimic în spațiul sinaptic, cantitatea de mediator descărcată fiind proporțională cu frecvența impulsurilor nervoase pe axon; spunem că în spațiul sinaptic informația privind intensitatea stimulului este *codificată în amplitudine*, aceasta fiind reprezentată de cantitatea de mediator descărcată.

La nivelul membranei postsinaptice, mediatorul se cuplează pe receptorii postsinaptici, se deschid canalele de sodium, membrana se depolarizează şi apare un potențial care se propagă pe membrana corpului neuronal sau pe dendrite. Spunem că informația este *codificată în amplitudine*, aceasta fiind reprezentată de potențialul local.

1.4. Informatica medicală

După această incursiune în teoria informației putem reveni la noțiunea centrală din informatică medicală şi anume *informația medicală*.

Ce este informația medicală şi când apare ea?

1.4.1. Date şi cunoştințe

Să încercăm să schițăm în cel mai simplificat mod actul medical primar şi anume vizita pacientului la medic. Poziția centrală în activitate medicală este ocupată de *pacient*. Fără pacient nu există medicină! Actorul principal al activității medicale este *medicul,* dar în activitatea medicală sunt implicate numeroase alte persoane care aparțin aşa-numitelor „*profesii aliate*". Dialogul medic-pacient începe uzual cu expunerea de către pacient a motivelor pentru care s-a prezentat la medic. Această descriere reprezintă transmiterea unor informații de la pacient către medic. Informațiile care se transmit sau se utilizează într-un act medical (sau ca urmare a unui act medical) reprezintă *informația medicală*. Dialogul este succedat de către examenul obiectiv al pacientului, medicul colectând astfel şi alte informații despre pacient. Să observăm că aceste informații au un caracter individual – sunt valabile pentru *acest pacient.* Aceste informații se numesc ***date***. Uzual paleta datelor se completează cu informații provenind şi din alte investigații (probe de laborator, explorări funcționale, radiografii etc.). Indiferent cât de complexe ar fi ca reprezentare, ele sunt „date", fiind caracteristice unui anumit individ.

În același timp, medicul utilizează şi alt fel de informații, numite *cunoştințe*. Acestea au un caracter general şi sunt acumulate în cursul pregătirii profesionale precum şi în experiența sa practică. Fără aceste cunoştințe informațiile sub formă de date nu pot fi interpretate (revenim la afirmația că valoarea utilă a informației depinde de receptor; practic, fără aceste cunoştințe receptorul datelor nu este „medic"). De aceea numeroşi autori numesc *informație* doar *datele interpretate*. Pentru a evita confuzia între termenul *informație* folosit la modul general şi *informație* pentru treapta de *date interpretate*, vom păstra termenul de *date interpretate* pentru acest nivel.

1.4.2. Ciclul elementar al informației medicale

Prin interpretarea datelor de către medic pe baza cunoştinţelor sale, este generată o nouă informație numită diagnostic. Pe baza diagnosticului, folosind din nou cunoştinţele sale, medicul stabileşte un plan terapeutic pe care îl aplică pacientului cu scopul de a îmbunătăţii starea pacientului. Urmărirea evoluției pacientului este însoțită de colectarea unor noi informații sub formă de date. Se observă că se închide un ciclu al fluxului informațional în activitatea medicală, numit „ciclul elementar al informației medicale" (figura 1.4.2).

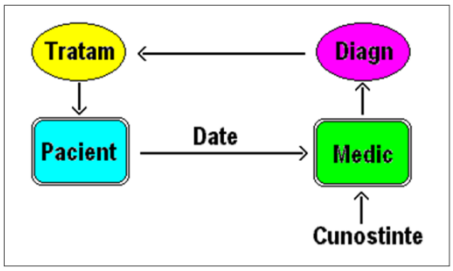


Fig. 1.4.2. Ciclul elementar al informației medicale

1.4.3. Tipuri de date

Informațiile culese despre starea pacientului, adică datele, pot îmbrăca diverse forme:

- *date calitative* – cu caracter descriptiv, aşa cum apar în anamneză
- *date numerice* – forma uzuală de prezentare a rezultatelor de laborator
- *grafice* – modul de înregistrare a evoluției în timp a unor mărimi biologice (ex.: semnalul ECG, EEG etc.)
- *sunete* – de ex. fonocardiograma; modul de prelucrare este asemănător cu cel al altor semnale
- *imagini* – radiografia, tomografia, ecografia etc.
- *imagini dinamice* – filme.

Modul de achiziție, stocare şi prelucrare este specific pentru fiecare tip de date şi în cadrul cursului nostru le corespund capitole separate.

1.4.4. Tipuri de cunoştinţe

Cunoştinţele pot fi de mai multe feluri:

- *cunoştinţe explicite* – care se pot formaliza, se pot exprima în propoziții, pot fi uşor transmise pe cale orală sau scrisă
- *abilități* sau *cunoştinţe tacite* (limba engleza - *skill*) – cele câştigate prin experiență practică (de ex. îndemânarea unui chirurg sau a unui dentist); nu pot fi transmise uşor.

1.4.5. Operații cu informații

Urmărind ciclul de viață al informației, din momentul generării sale până în momentul utilizării, observăm că informația suferă o serie de operații:

- *achiziția (colectarea)* – presupune mijloace specifice tipului de informație
- *stocarea* – baze de date, respective baze de cunoştinţe
- *transmitere* – căi, procedee

- *prelucrare* – cu o largă paletă de metode specifice, pentru a extrage elementele esențiale în vederea interpretării şi utilizării
- *protecție* – măsurile ce se impun pentru asigurarea integrității informației stocate sau transmise, precum şi a confidențialității acesteia
- *interpretare/utilizare* – pasul final, în care informația este integrată în acțiunile specifice nivelului.

1.5. Capitolele informaticii medicale şi structura cursului

1.5.1. Clasificarea informației medicale pe nivele structurale

În ciclul elementar al informației medicale prezentat mai sus am luat în considerare informațiile care apar în activitatea medicală curentă, la nivelul individului, numit pacient. Totuşi fenomenele care se petrec în materia vie (legate de starea de sănătate a pacientului) privesc deseori nivele infraindividuale, pornind de la nivelul molecular sau celular, urcând prin nivelul de țesut, organ sau sistem până la nivelul întregului organism sau nivelul individual.

Pe de altă parte, activitatea medicală este organizată în unități care prestează servicii pentru populație, deci putem urmări fluxul informațional şi la nivel supraindividual, de comunitate. Corespunzător acestor nivele structurale avem diferite discipline biomedicale precum şi diferite capitole corespunzătoare ale informaticii medicale.

1.5.2. Bioinformatica – relații cu alte discipline

Bioinformatica este un capitol de sine stătător al informaticii medicale. Însă, prin specificul obiectului său de a urmări modul în care este reprezentată informația în structurile vii, obiectul său este strâns legat de o serie de discipline privind structura şi funcția structurilor din materia vie. Astfel, sunt necesare cunoştințele de biofizică, biochimie, biologie celulară şi moleculară precum şi genetică.

Din punct de vedere al abaterilor formale, legătura cu teoria probabilităților, biostatistica, teoria sistemelor şi chimia cuantică este evidentă. De asemenea, se folosesc diverse limbaje de programare – R, PERL, C++, Java etc.

După cum vedem, se poate afirma – pe bună dreptate – că bioinformatica este una dintre cele mai complexe discipline, pregătirea specialiştilor în acest domeniu fiind cel mai adesea inclusă în categoria „studii avansate”.

2. Introducere în biofizică

2.1. Obiectul şi capitolele biofizicii

2.1.1. Ştiinţe interdisciplinare

A. Evoluţia cunoaşterii umane a consacrat de-a lungul timpului conturarea unor domenii solide, bine definite mai ales în conţinut decât în definiţii formale. Astfel şi-au stabilit o poziţie solidă diferite ştiinţe: matematica, fizica, chimia, biologia etc. Ele au nu numai obiect diferit ci şi mod de abordare specific, considerat adesea chiar „mod de gândire". Însă complexitatea naturii şi modul elaborat al gândirii umane au şters graniţele dintre ştiinţele clasice, fertilizând un teritoriu numit astăzi ştiinţe interdisciplinare, ce au cunoscut o dezvoltare deosebită în secolul XX şi care s-au dovedit a fi foarte prolifice.

B. În acest context putem privi şi intersecţia între două domenii gigant – fizica, respectiv biologia. Mai multe discipline de graniţă şi-au găsit originea în această intersecţie: biofizica, bionica, biocibernetica, biotehnologia, fizica medicală etc., fiecare cu obiect şi metode specifice. Putem astfel preciza obiectul câtorva dintre aceste ştiinţe de graniţă, cele mai bine conturate şi dezvoltate.

Biofizica – este disciplina ce se ocupă cu studiul fenomenelor fizice care se petrec la nivelul materiei vii.

Bionica – se ocupă cu aplicarea în tehnică a unor „soluţii" din natură (imitarea unor procese din materia vie). Ex.: forma elicopterului, profilul aripilor de avion, ecolocaţia etc.

Biocibernetica – studiază mecanismele de reglare, control şi comandă în materia vie precum şi integrarea diferitelor nivele informaţionale în sistemele biologice.

Biotehnologia – utilizarea organismelor vii sau bioproceselor în inginerie, tehnologie sau medicină ex.: inginerie genetică, culturi de celule sau de ţesuturi, etc.

Fizica medicală – aplicaţii ale fizicii în medicină, pentru diagnostic (toată aparatura medicală radiografia, ecografia etc.) sau tratament (cu ultrasunete, terapia prin radiaţii etc.).

2.1.2. Capitolele biofizicii

Există mai multe variante de împărţire a biofizicii pe capitole:

A. În funcţie de fenomenul fizic urmărit de ex.: biomecanică, bioelectricitate, biomagnetism, fenomene optice în biologie etc.

B. În funcţie de nivelul structural abordat astfel avem:

a) *biofizica moleculară* – care se ocupă cu studiul proprietăţilor moleculelor componente ale materiei vii,

b) *biofizica celulară* – în care se urmăresc fenomenele fizice la nivelul celular, inclusiv particularităţile de abordare a fenomenelor fizice din materia vie în general; în acest capitol sunt incluse: termodinamica biologică, fenomenele de transport – inclusiv

transportul transmembranar, respectiv bioelectrogeneza – generarea fenomenelor electrice la nivelul membranelor celulare,

c) *biofizica sistemelor complexe* – aspecte fizice ale unor sisteme: aspecte mecanice ale sistemului osteomuscular, aspecte de dinamica fluidelor în sistemul circulator, aspecte specifice în organele de simţ mecanismul văzului, auzului etc.,

d) *biofizica ambientală*, numită frecvent şi interacţiunea factorilor fizici cu materia vie, capitol dedicat variatelor interacţiuni cu diverşi factori fizici: presiune, curent electric, unde mecanice - vibraţii, ultrasunete, unde electromagnetice toată gama, de la unde radio şi microunde la radiaţii infraroşii, ultraviolete, X şi gamma, radiaţii corpusculare neutroni etc.

C. Vom prezenta doar câteva noţiuni fundamentale din capitolele de biofizică moleculară şi celulară utile pentru înţelegerea fenomenelor ce vor fi prezentate în capitolele următoare.

Din biofizica moleculară vom prezenta pe scurt:
- structura atomului
- atomul de carbon
- structura moleculară
- forţe intermoleculare
- molecula de apă
- soluţii
- noţiuni generale despre pH.

Din biofizica celulară ne vom opri la:
- noţiuni de termodinamică biologică
- forţe termodinamice
- procese cuplate
- transport transmembranar.

2.2. Structura atomului

2.2.1. Proprietăţi

A. Atomul are dimensiuni de ordinul 10^{-10}m (10^{-10}m = 1Å Ångstrom). Masa se exprimă în unităţi atomice de masă sau daltoni (1u.a.m = 1u = 1Da = 1,66 × 10^{-27}kg). Masa relativă a atomului este numărul care arată de câte ori este mai greu atomul respectiv faţă de u.a.m., de ex.: masa relativă a atomului de carbon ^{12}C este 12.

B. Atomul este neutru din punct de vedere electric.

C. Atomul este format din nucleu şi nor electronic.

2.2.2. Nucleul atomic

A. Nucleul atomic are dimensiuni de ordinul 10^{-15}m şi concentrează practic masa întregului atom.

B. Nucleul este încărcat electric cu sarcină pozitivă. Nucleul este format din protoni şi neutroni. Protonii sunt particule elementare încărcate pozitiv cu sarcina +1e şi masa aproximativ 1u; neutronii sunt particule elementare neutre electric, cu masa aproximativ 1u.

C. Nucleul este caracterizat prin două numere:

- *numărul de ordine* Z (sau *număr atomic*), care reprezintă numărul de protoni din nucleu şi determină poziţia atomului în tabelul periodic al elementelor. Sarcina nucleului este +Ze, unde „e" este sarcina electrică elementară (1e = 1,6 × 10^{-19}C, C = coulomb)

- *numărul de masă* A, care reprezintă numărul total de protoni şi neutroni din nucleu. Cum atât masa protonului cât şi a neutronului sunt apropiate de 1u, numărul de masă reprezintă aproximativ masa nucleului exprimată în u.

D. Simbolic un nucleu se notează $_{Z}\mathbf{X}^{\mathbf{A}}$, unde X este simbolul chimic al atomului iar Z şi A, ca indici inferiori, respectiv superior, sunt numărul de ordine, respectiv numărul de masă. (A si Z se pot scrie şi ambii de aceeaşi parte a simbolului elementului).

2.2.3. Norul electronic

A. Nucleul este înconjurat de un nor electronic, alcătuit din Z electroni, fiecare cu sarcina negativă elementară.

a) Norul electronic este împărţit în nivele sau straturi. Un strat electronic este caracterizat prin numărul cuantic **n**, numit *număr cuantic principal*, care determină energia E_n şi dimensiunea norului electronic r_n;

- valori posibile **n = 1, 2, 3, ...** şi corespund nivelelor energetice ale straturilor notate K, L, M, ... ;
- energia nivelului $\mathbf{E_n \sim -1/n^2}$, iar distanţa medie (~ raza orbitei) $\mathbf{r_n \sim n^2}$.

Observaţie: energia electronului legat în atom este negativă, electronii din apropierea nucleului sunt puternic legaţi, cei periferici sunt slab legaţi; când $n \to \infty$, $E \to 0$ şi electronul devine liber. Pentru smulgerea unui electron din atom este necesară o energie egală cu energia sa de legătură, iar atomul devine un ion pozitiv.

Ex.: pentru Z = 1, E_1 = - 13, 6eV iar r_1 = 0,53Å (1eV = 1,6 × 10^{-19}J, J = Joule).

b) Fiecare strat (nivel) este împărţit în subnivele; un subnivel într-un nivel este caracterizat prin numărul cuantic **l** numit *număr cuantic orbital* sau azimutal, care determină forma norului electronic;

- valori posibile **l = 0, 1, ... , n,** în total n valori, corespunzând subnivelelor notate în ordine s, p, d, f. Energia subnivelelor creşte cu l, dar mai slab decât variaţia cu n; un nivel are n subnivele;
- forma norului pentru:
  - s = 0 - formă sferică
  - s = 1 - formă bilobară
  - s = 2 - formă tetralobară etc.

c) Fiecare subnivel este împărţit în orbitali; un orbital într-un subnivel este caracterizat prin *numărul cuantic magnetic* **m**, care determină orientarea în spaţiu a norului electronic;

- valori posibile **m = -l, ..., -1, 0, +1, ..., +l**, în total 2l+1 valori. Toţi orbitalii unui subnivel au aceeaşi energie şi se numesc orbitali „degeneraţi". Subnivelele s (l = 0) au un singur orbital (sferic); subnivelele p (l = 1) au trei orbitali (bilobari), orientaţi pe cele trei direcţii din spaţiu: p_x, p_y, p_z.

B. Notația simbolică a unui orbital cuprinde numărul nivelului (n) urmat de simbolul subnivelului (s, p, d, f), la care se adaugă un indice privind orientarea spațială determinată de numărul cuantic magnetic; (în cazul orbitalilor s, de formă sferică acest indice nu are sens).

Deci un orbital într-un atom este caracterizat prin 3 numere cuantice:
- nr cuantic principal **n** (n = 1, 2, 3..., pentru nivelele K, L, M, ...)
- nr cuantic orbital **l** (l = 0, 1, ..., n-1, pentru subnivelele s, p, d, f)
- nr cuantic magnetic **m** (m = -l, ..., 0, ..., +l).

C. Un electron într-un atom este caracterizat prin 4 numere cuantice:
- 3 numere cuantice n, l, m ce determină orbitalul pe care se găseşte
- numărul cuantic magnetic de *spin* **s**, care defineşte electronul pe orbital
- valori posibile: s = +1/2, -1/2, în total 2 valori.

D. Principiul lui Pauli

Enunț: Într-un atom nu pot exista mai mulți electroni cu aceleaşi valori pentru cele 4 numere cuantice.

Consecință: pe un orbital pot exista maxim 2 electroni, cu spin opus.

2.2.4. Structura norului electronic – principii

A. Atomii diferă între ei prin structura norului electronic, care determină proprietățile chimice ale atomului şi principalele proprietăți fizice.

Structura norului electronic poate fi analizată ierarhic, pornind de la cel mai simplu atom – atomul de hidrogen, care are un singur electron şi urmărind în continuare structura pentru atomii cu mai mulți electroni.

B. În completarea starturilor electronice pentru atomi vom lua în considerare următoarele principii şi reguli:

a) principiul energiei minime nivelele subnivelele orbitalii se completează începând cu orbitalii de cea mai joasă energie;

b) principiul lui Pauli pe un orbital încap maxim doi electroni, cu spin diferit;

c) regula lui Hund în cazul orbitalilor degenerați se completează întâi fiecare orbital cu câte un electron cu acelaşi spin, apoi cel de-al doilea electron pe fiecare orbital.

2.2.5. Structura norului electronic al elementelor

Cu principiile prezentate mai sus putem reprezenta schematic structura norului electronic al elementelor de la începutul tabelului periodic al elementelor.

În figura 2.2.5.a sunt prezentate simbolic structurile norului electronic pentru elementele cu Z până la 13.

$_{1}$**H** - $1s^1$ $_{2}$**He** - $1s^2$ $_{3}$**Li** - $1s^2\ 2s^1$

$_{4}$**Be** - $1s^2\ 2s^2$ $_{5}$**B** - $1s^2\ 2s^2\ 2p_x^1$

$_{6}$**C** - $1s^2\ 2s^2\ 2p_x^1\ 2p_y^1$, $_{7}$**N** - $1s^2\ 2s^2\ 2p_x^1\ 2p_y^1\ 2p_z^1$,

$_{8}$**O** - $1s^2\ 2s^2\ 2p_x^2\ 2p_y^1\ 2p_z^1$, $_{9}$**F** - $1s^2\ 2s^2\ 2p_x^2\ 2p_y^2\ 2p_z^1$,

$_{10}$**Ne** - $1s^2\ 2s^2\ 2p_x^2\ 2p_y^2\ 2p_z^2$, $_{11}$**Na** - $1s^2\ 2s^2\ 2p_x^2\ 2p_y^2\ 2p_z^2\ 3s^1$,

$_{12}$**Mg** - $1s^2\ 2s^2\ 2p_x^2\ 2p_y^2\ 2p_z^2\ 3s^2$, $_{13}$**Al** - $1s^2\ 2s^2\ 2p_x^2\ 2p_y^2\ 2p_z^2\ 3s^2\ 3p_x^1$

Fig. 2.2.5.a. Structura norului electronic al elementelor de la începutul tabelului periodic

În fig. 2.2.5.b. sunt prezentate pozițiile relative ale nivelelor şi subnivelelor energetice.

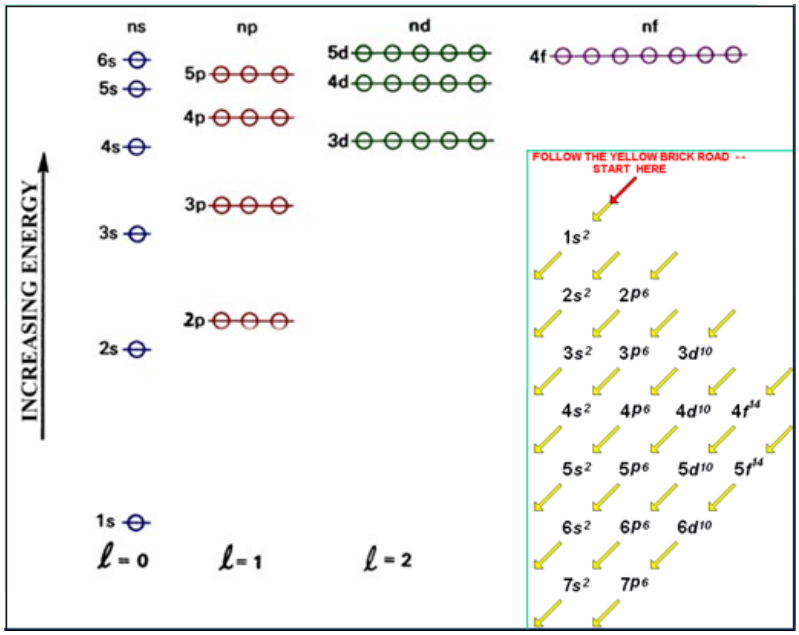


Fig. 2.2.5.b. Structura norului electronic şi poziția relativă a subnivelelor energetice

2.3. Atomul de carbon

2.3.1. Structura norului electronic

A. Atomul de carbon ocupă poziția centrală în chimia organică, numeroase proprietăți ale moleculelor din materia vie fiind mai uşor înțelese dacă înțelegem configurația norului electronic al atomului de carbon.

Din cei 6 electroni, 2 sunt pe stratul K, aproape de nucleu, pe care nu îl părăsesc în nici un proces biochimic, deci îl neglijăm în comentariile ce urmează.

B. Cei 4 electroni de pe stratul L sunt situați în stare fundamentali în structura **$2s^2$, $2p_x^1$, $2p_y^1$**.

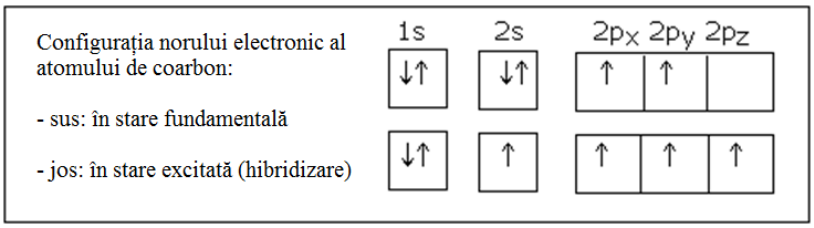


Fig. 2.3.1. Structura norului electronic al atomului de carbon

Aceasta explică comportamentul bivalent în unele situații, de exemplu în molecula de CO, prin completarea orbitalilor $2p_x$ și $2p_y$ care erau incompleți.

2.3.2. *Hibridizarea*

A. Diferența de energie între subnivelul 2s și 2p este foarte mică, astfel încât un electron de pe 2s sare pe orbitalul liber $2p_z$, având acum 4 orbitali cu câte un electron fiecare, cu același spin (conform regulii lui Hund). Se explică astfel caracterul uzual tetravalent al atomului de carbon.

Saltul electronului de pe 2s pe $2p_z$ se numește hibridizare.

B. Hibridizarea este însoțită de o reorientare în spațiu a orbitalilor. Aceasta se poate realiza în trei feluri:

a) *Hibridizare* sp^3: toți orbitalii se reorientează simetric în spațiu, cu densitate maximă orientată spre vârfurile unui tetraedru și toți au aceeași energie, caz în care atomul de carbon are cele 4 valențe echivalente și le poate angaja în diferite legături chimice.

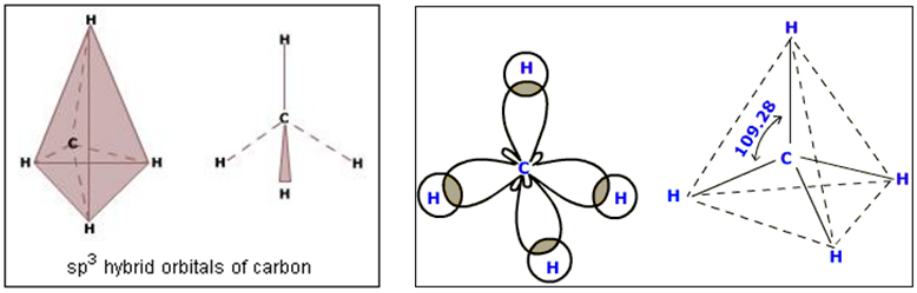


Fig. 2.3.2.a. Hibridizarea sp^3

Prin această orientare în spațiu a valențelor se pot explica și o serie de proprietăți, cum ar fi chiralitatea izomeria sterică, sau izomeria optică, în funcție de atomii cu care se realizează legăturile.

b) *Hibridizarea* sp^2 . În cazul în care un electron de pe un orbital, (ex.: $2p_x$), era angajat într-o legătură chimică, (de exemplu cu un alt atom de carbon), reorientarea va cuprinde numai orbitalul s și ceilalți 2 orbitali p (fig. 2.3.2.b).

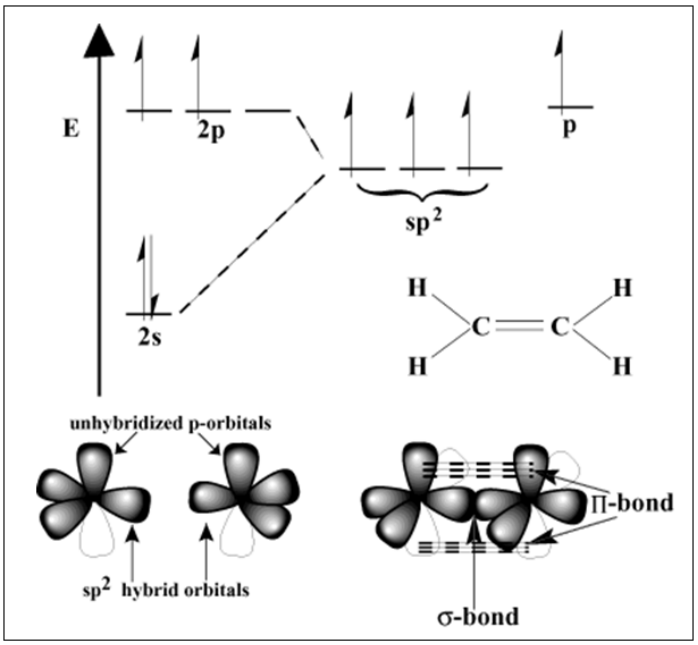


Fig. 2.3.2.b. Hibridizarea sp^2

În acest caz poate să apară şi o a doua legătură între doi atomi, prin întrepătrunderea intersecția norului electronic ai orbitalilor care nu erau orientați de-a lungul axei dintre atomi ci într-un plan perpendicular. Evident, cele două legături nu sunt echivalente; prima, generată la intersecția orbitalilor de-a lungul axei este puternică şi se numeşte legătură **σ**, iar a doua este mai slabă şi se numeşte legătură **π**. (Comparativ, pentru legătura simplă C-C, energia legăturii este 347 kJ/mol, iar a legăturii **π** din legătura dublă este 263kJ/mol).

Important de remarcat că ceilalți orbitali se orientează simetric față de norul format.

c) *Hibridizarea sp*. În cazul în care ambii electroni de pe 2p sunt angajați în legături chimice, reorientarea se face între orbitalul s şi orbitalul $2p_z$ rămas gol.

Într-o astfel de situație apare şi legătura triplă, în care apare o legătură **σ** şi două legături **π**, (mai slabe în medie decât când era doar una).

C. Orientarea norului electronic în moleculele organice (în special proteine) este un subiect important, prin aceasta explicându-se proprietățile lor. Există software dedicat pentru astfel de analize.

2.4. Structura moleculei. Legături chimice

2.4.1. Stabilitatea atomilor

A. Atomii au tendința de a avea un nor electronic complet. În funcție de structura norului electronic putem estima gradul de stabilitate şi proprietățile diverselor clase de atomi. Astfel, atomii cu norul electronic complet sunt foarte stabili (grupa 18) - gazele nobile (inerte).

B. Atomii cu puțini electroni pe ultimul strat (electroni de valență) au tendința de a-i ceda mai uşor (caracter electropozitiv), în timp ce atomii cărora le lipsesc puțini electroni pentru completarea subnivelului periferic au tendința de a acapara electroni (caracter electronegativ). Cu cât aceste tendințe sunt mai puternice, atomii respectivi sunt mai puțini stabili (mai reactivi).

2.4.2. Legătura covalentă

A. Cea mai evidentă manifestare a tendințelor atomilor către configurația electronică stabilă apare prin formarea legăturilor chimice, iar dintre acestea, cea mai frecventă (şi mai generală) formă este legătura covalentă.

Legătura covalentă se realizează prin punerea în comun a electronilor de pe orbitalii incompleți (cu câte 1 electron). Orbitalul nou format se numeşte orbital molecular şi se redistribuie spațial între cei doi atomi, existând posibilitatea – prin chimie cuantică - de a calcula densitatea norului electronic nou format.

B. Legătura covalentă nepolară

În cazul în care legătura covalentă se realizează între atomi identici, norul electronic molecular va fi simetric distribuit peste ambii atomi, centrul sarcinilor pozitive (aflat la mijlocul distanței între cele două nuclee) va coincide cu centrul sarcinilor negative; molecula formată este nepolară (exemple: H_2, Cl_2, figura 2.4.2. sus). Schematic perechea de electroni puşi în comun se notează cu o linie „-”, cu semnificația de legătură simplă (o singură pereche de electroni); în mod uzual electronii neparticipanți nici nu se reprezintă.

C. Legătura covalentă polară

În cazul în care atomii sunt diferiți, atomul mai electronegativ atrage norul electronic mai mult spre sine şi va deveni un centru al sarcinilor negative, în timp ce atomul mai slab electronegativ va rămâne parțial denudat de nor electronic, devenind un centru al sarcinilor pozitive. Molecula obținută este o moleculă polară, este caracterizată printr-un moment de dipol (μ) şi se va orienta de-a lungul liniilor de câmp în cazul plasării într-un câmp electric. De asemenea, fiind un dipol, se va implica într-o serie de fenomene electrice cu caracter molecular. Exemple: HCl, prezentat în figura 2.4.2. jos, H_2O etc.

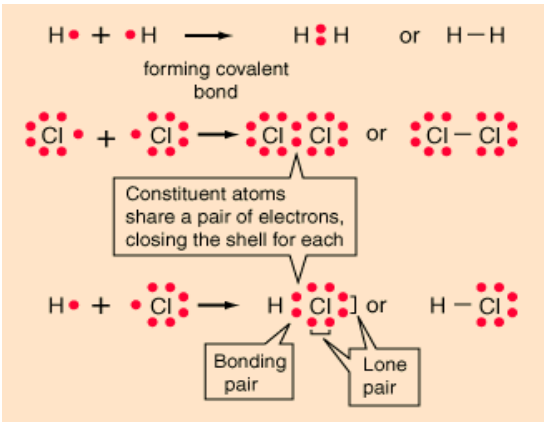


Fig. 2.4.2. Legătura covalentă polară şi nepolară

2.4.3. Energia de legătură

A. Energia de legătură a unei legături chimice este definită ca energia necesară pentru ruperea sa (deci este o energie negativă) şi este egală cu energia eliberată la formarea moleculei. Se exprimă uzual în kJ/mol (se mai foloseşte kcal/mol).

B. Un grafic al energiei potențiale a unui sistem de doi atomi este prezentat în figura 2.4.3. şi poartă numele de groapă de potențial. Distanța între atomi pentru energia potențială minimă se numeşte distanța de echilibru.

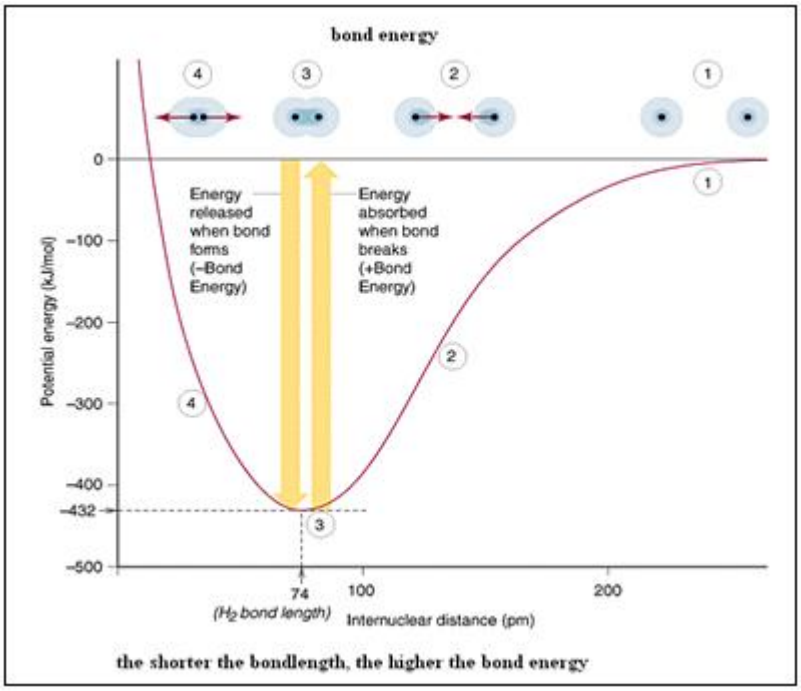


Fig. 2.4.3. Energia de legătură

C. Atomii nu sunt ficşi la distanța de echilibru ci execută mişcări de oscilație vibrație în jurul poziției de echilibru. Amplitudinea vibrațiilor creşte cu temperatura. Nivelele energetice de vibrație sunt cuantificate. Fiecare moleculă are un spectru de vibrație specific, cu benzi situate uzual în regiunea infraroşie a spectrului.

D. Stabilitatea unei molecule depinde de energia de legătură: cu cât energia de legătură este mai mare (la formare s-a eliberat mai multă energie), cu atât molecula este mai stabilă.

E. Există o relație între distanța de echilibru şi energia de legătură pentru energii de legătură mai mari, forțele de atracție sunt mai puternice şi distanța de echilibru lungimea legăturii este mai mică.

2.4.4. Legătura ionică

A. Un caz limită al legăturii covalente polare este cazul în care elementul mai electronegativ preia integral electronul pus în comun, devenind iar negativ, iar celălalt atom a devenit un ion pozitiv. Ex.: NaCl, MgO (figura 2.4.4.a).

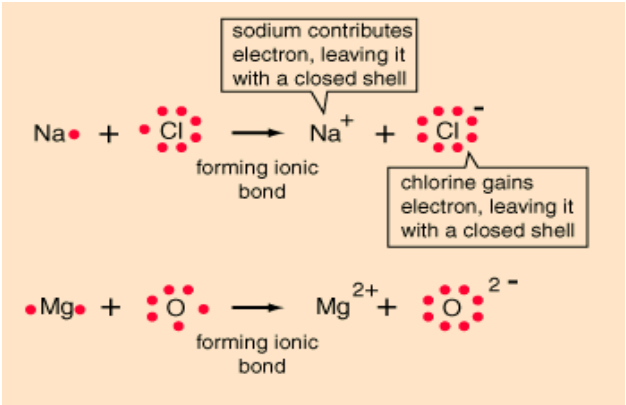


Fig. 2.4.4.a. Legătura ionică

B. Între cei doi ioni se exercită o forță electrostatică coulombiană, iar energia de legătură va fi chiar energia potențială corespunzătoare acestei interacțiuni (figura 2.4.4.b)

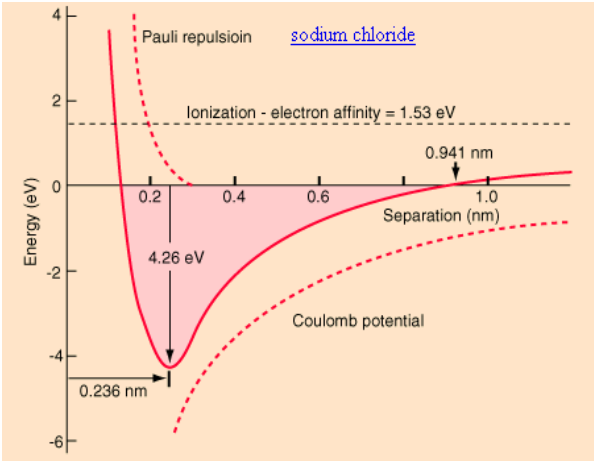


Fig. 2.4.4.b. Energia legăturii ionice

C. În cazul legăturii ionice, noțiunea de moleculă devine formală, în structuri avem doar ioni independenți, pozitivi şi negativi. Aceasta este situația şi în cristale, (de ex.: NaCl) şi în soluție; să facem totuşi observația că ionii în soluție sunt hidratați, (moleculele de apă fiind polare, sunt atrase cu capătul de sarcină opusă către ion).

2.5. *Forțe intermoleculare*

Moleculele din structuri interacționează între ele. Aceste interacțiuni sunt mai slabe decât legăturile chimice, însă existența forțelor intermoleculare determină o serie de proprietăți importante ale moleculelor, ex.: solubilitatea.

Ele pot fi:
- legătura de hidrogen
- forțe Van der Waals
- forțe de dispersie.

2.5.1. Legătura de hidrogen

A. Hidrogenul participant în legăturile covalente polare este adesea victima unei interacţiuni cu un atom mai electronegativ, rămânând parţial privat de nor electronic şi un centru de sarcină pozitivă. În aceste condiţii el poate fi atras de un nor electronic complet al unui orbital, creat de o pereche de electroni neparticipanţi la vreo legătură chimică (deci nor dens în jurul unui atom electronegativ - centru de sarcină negativă).

B. Legătura de hidrogen este definită ca interacţiune de natură electrostatică între hidrogenul unei molecule cu electronii neparticipanţi ai unui atom din altă moleculă, de obicei un atom de oxigen sau azot.

C. Energia de legătură în cazul punţilor de hidrogen este de 10 ori mai slabă decât a legăturilor covalente (4,5 kcal/mol, faţă de 110 kcal/mol în legătura O-H). Exemple: H_2O, NH_3, A-T/G-C etc. (figura 2.5.1).

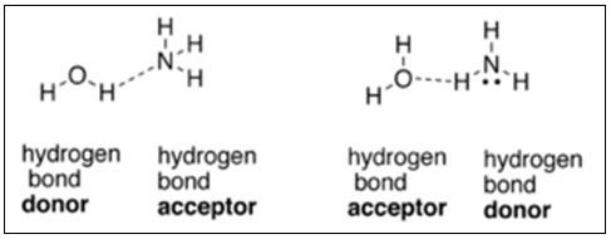


Fig. 2.5.1. Punţi de hidrogen

2.5.2. Forţe Van der Waals

A. Forţele van der Waals sunt forţe de atracţie slabe, ce se manifestă doar pe distanţe foarte mici, scăzând foarte repede cu distanţa dintre molecule. Ele pot fi datorate interacţiunilor de origine electrostatică între dipolii moleculari.

B. Astfel putem avea:

– interacţiuni dipol - dipol, care apar între centrul sarcinilor pozitive ale unei molecule polare şi centrul sarcinilor negative al altei molecule polare;
– interacţiuni dipol - dipol indus, care pot implica şi molecule nepolare, dar care, sub acţiunea câmpului electric al unei molecule puternic polare, îşi deformează norul electric şi apare un dipol indus care interacţionează cu molecula polară.

2.5.3. Forţe de dispersie

Forţele de dispersie sunt şi mai slabe ca forţele Van der Waals, acţionează doar la distanţe foarte mici şi cresc odată cu masa moleculară. Le luăm în considerare doar dacă celelalte forţe sunt toate mici.

2.6. Molecula de apă

2.6.1. Structura moleculară

A. Apa este un component esenţial al materiei vii, având o serie de proprietăţi care trebuie luate în considerare pentru a înţelege o suită de fenomene din lumea vie.

B. Structura moleculară a apei este bine cunoscută, H_2O. Norul electronic este deplasat către oxigen, care devine centru de sarcini negative, centrul sarcinilor pozitive fiind la semidistanța între atomii de hidrogen. Datorită respingerii electrostatice dintre nucleele de hidrogen, unghiul între valențe creşte de la 90^o la 105^o, iar orbitalii cu electroni neparticipanți ai oxigenului se reorientează spre vârfurile unui tetraedru (neregulat, nu identic cu norul carbonului, dar asemănător). În figura 2.6.1. sunt prezentate schematic ideile redate aici, formarea legăturilor covalente O-H, unghiul între valențele O-H, tetraedrul cu poziția atomilor de hidrogen şi densitățile maxime ale norului electronilor neparticipanți.

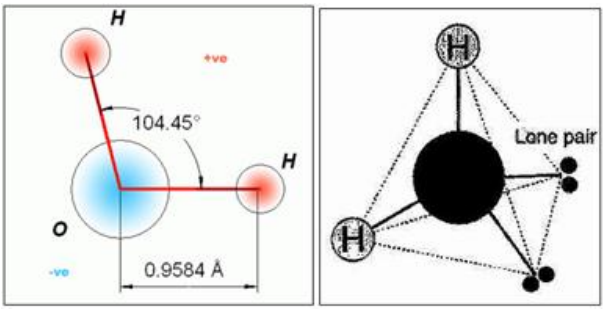


Fig. 2.6.1. Structura moleculei de apă

2.6.2. Legăturile de hidrogen ale moleculei de apă

A. O moleculă de apă poate prezenta 4 legături de hidrogen

– două legături, stabilite de cei doi atomi de hidrogen, care se îndreaptă, fiecare, spre o altă moleculă de apă, mai precis către electronii neparticipanți ai altei molecule de apă;

– alte două legături prin cei doi lobi corespunzători orbitalului 2p care hibridizează cu 2s, formând un nor cu densități maxime către celelalte două vârfuri ale tetraedrului.

B. O reprezentare sugestivă, folosită frecvent în chimia organică şi biochimie, este prezentată în figura 2.6.2, în care cu culoare închisă este redat atomul de oxigen, iar cu culoare deschisă atomul de hidrogen. Sunt evidențiate cele 4 legături posibile.

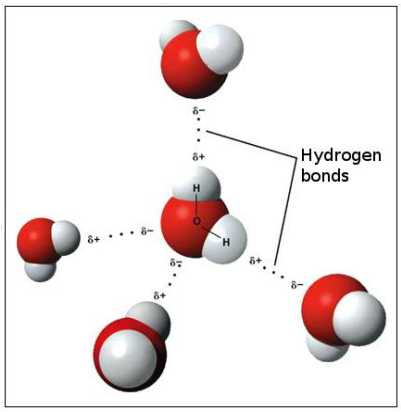


Fig. 2.6.2. Legăturile de hidrogen ale moleculei de apă

2.6.3. Proprietățile apei

A. Prezența legăturilor de hidrogen face ca apa să ocupe un loc excepțional comparativ cu alte molecule, favorizând poziția sa privilegiată ca şi component esențial al materiei vii. Fără a intra în detalii explicative, prezentăm în continuare o listă a proprietăților apei, în care, în bună parte, explicațiile invocă prezența punților de hidrogen:

- gheața are structură cristalină hexagonală, cu spații libere, având densitatea mai mică decât apa lichidă,
- la 0°C numai 15% din legăturile de hidrogen se rup; apa lichidă are structură cvasicristalină, formată în special din trimeri şi dimeri,
- densitatea apei este maximă la 4°C,
- tensiunea superficială a apei este foarte mare (cca. 72 N/m, față de cca. 22 N/m pentru alcool),
- căldura specifică (implicit capacitatea calorică) mare, acumulând o parte din energie pentru ruperea unor legături de hidrogen, deci ca energie potențială şi nu ca energie cinetică de vibrație a moleculelor; inerția termică este considerată ca unul din factorii cheie în evoluția materiei vii (c = 4,18 kJ/kg.grd = 1 kcal/kg.grd),
- temperatura de topire t_t = 0°C, temperatura de fierbere t_f = 100°C (față de alcool: t_t = -38°C, t_f = 78°C),
- conductibilitatea termică ridicată (~ 0,0013 cal/cm^2.grd),
- căldurile latente foarte mari: căldura latentă de topire λ_t ~ 80 kcal/kg = 335 kJ/kg, căldura latentă de fierbere λ_f ~ 580 kcal/kg = 2258 kJ/kg,
- conductibilitatea electrică foarte mică (rezistivitate mare, $\sigma \sim 11.10^{-6}\ \Omega^{-1}.m^{-1}$),
- constanta dielectrică relativă foarte mare (ε = 80), datorită efectului dipolilor moleculari,
- proprietăți optice: transparență, indice de refracție n = 4/3, nu absoarbe în domeniul vizibil, absoarbe moderat radiațiile UV şi este opacă la radiațiile IR.

B. Vâscozitatea – este o proprietate ce reflectă gradul de frecare internă între molecule în cazul curgerii lichidului. Vâscozitatea scade cu temperatura. În cazul apei se constată o scădere aproximativ liniară de la 4°C la 35 °C , apoi o scădere bruscă în intervalul 35 - 40°C şi din nou o scădere liniară până la fierbere. Aceasta datorită faptului că în intervalul 35 - 40°C are loc ruperea masivă a unor punți de hidrogen,nu mai rămân în soluție trimeri (grupe de trei molecule de apă) şi mulți dimeri se desfac. Temperatura de 37°C se mai numeşte al doilea punct de topire al apei şi nu întâmplător animalele homeoterme au ca temperatură de echilibru tocmai această valoare.

Proprietățile enumerate mai sus, chiar dacă nu au fost explicate în detaliu, joacă un rol esențial în înțelegerea unui mare număr de procese biologice şi vom face referiri specifice unde este cazul.

2.7. Soluții

2.7.1. Sisteme disperse

A. În natură nu ne întâlnim cu sisteme pure, formate dintr-un singur fel de molecule, ci cu sisteme formate din mai multe tipuri de molecule. Aceste sisteme poartă numele generic de sisteme disperse.

Repartiția spațială a tipurilor de molecule poate fi uniformă (sisteme omogene) sau neuniformă (sisteme heterogene). Pentru moment vom lua în considerare doar sistemele omogene.

B. Compoziția sistemelor disperse

În general un sistem dispers are două componente principale, numite *faze:*

- o fază continuă, numită *solvent*, alcătuind componenta majoritară a sistemului,
- o fază discontinuă discretă, numită *solut*, reprezentată de substanța dizolvată.

C. Cel mai adesea considerăm cele două faze ca fiind reprezentate de starea de agregare lichidă. Totuşi, la modul general, putem defini sisteme disperse cu toate combinațiile posibile pentru solvent şi solut.

D. Clasificarea sistemelor disperse după diametrul particulelor solutului.

a) *soluții moleculare*, cu diametrul sub10 Å; cum de obicei masa moleculară este proporțională cu volumul, deci cu puterea a treia a diametrului, uzual se consideră în categoria soluții moleculare, soluțiile care au pentru substanța dizolvată masa moleculară M<1000,

b) *soluții coloidale*, cu diametrul între 10 şi 1000 Å şi masa moleculară peste 10^3, dar sub 10^6,

c) *dispersii* – medii cu particule cu diametrul 1000 Å sau M>10^6.

E. O altă clasificare posibilă ia în considerare procesul de disociere; astfel soluțiile pot fi:

a) electroliți – în cazul în care moleculele disociază să specificăm aici că electroliții pot fi la rândul lor:

i. electroliți tari, când disocierea este totală este cazul tuturor sărurilor, dar şi acizii şi bazele tari;
ii. electroliți slabi, când disocierea este parțială, acizi sau baze slabe în aceste situații trebuie definită şi o constantă de disociere, ca raportul între numărul moleculelor disociate şi numărul total de molecule dizolvate.

b) ne-electroliți – când substanțele dizolvate sunt formate din molecule care nu disociază, ex.: glucoză, uree etc.

2.7.2. Concentrații

A. O caracteristică esențială a soluțiilor este concentrația acestora, exprimată ca măsură a proporției moleculelor substanței dizolvate în soluție.

Există mai multe modalități de a exprima concentrația unei soluții:

a) Concentrația procentuală – exprimă cantitatea de substanță dizolvată, exprimată în grame, la 100 ml soluție;

b) Concentrația molară – exprimă numărul de moli de substanță dizolvată în 1l soluție (1M = 1 mol = cantitatea de substanță, exprimată în grame, numeric egală cu masa moleculară exprimată în u.a.m.; 1 mol are N_A molecule, N_A este numărul lui Avogadro N_A = 6, 023.10^{23} molecule/mol);

c) Concentrația normală – exprimă numărul de echivalenți gram de substanță dizolvați în 1l de soluție (1 echivalent-gram = l mol/z, unde z este numărul de sarcini electrice de un semn, ce se obține la disocierea unei molecule).

B. Relații

$$C(M/l) = m(g)/V(l).M \qquad (2.7.2)$$

unde C este concentrația, în mol/litru, V = volumul soluției în l, M este masa moleculară

2.8. pH-ul soluțiilor

2.8.1. Disocierea electroliților

A. Electroliții disociază în soluție. Disocierea poate fi totală sau parțială. În cazul disocierii parțiale se defineşte constanta de disociere k.

B. Acizi

Substanțele care prin disociere pun în libertate protoni ioni H^+ se numesc acizi. Acizii tari disociază total, acizii slabi disociază parțial. Ex.: HCl, H_2CO_3

$$AH \rightarrow A^- + H^+ \quad \text{(acizi tari), sau} \qquad (2.8.1.a)$$

$$AH \leftrightarrow A^- + H^+ \quad \text{(acizi slabi)} \qquad (2.8.1.a')$$

C. Baze

Substanțele care prin disociere pun în libertate ioni OH se numesc baze (substanțe alcaline). Bazele tari disociază total, bazele slabe disociază parțial. Ex.: NaOH, NH_4OH.

$$BOH \rightarrow B^+ + OH^- \quad \text{(baze tari), sau} \qquad (2.8.1.b)$$

$$BOH \leftrightarrow B^+ + OH^- \quad \text{(baze slabe)} \qquad (2.8.1.b')$$

D. Disocierea apei

Molecula de apă disociază parțial în soluție eliberând un proton de H^+ şi un ion oxidril (hidroxil) OH^-.

$$H_2O \leftrightarrow H^+ + OH^- \qquad (2.8.1.c)$$

E. Constanta de echilibru

În cazul disocierii parțiale, putem explica legea acțiunii maselor constanta de echilibru se numeşte constantă de disociere. Iată, pentru acizii slabi şi bazele slabe.

$$AH \leftrightarrow A^- + H^+ \qquad k_a = [A^-].[H^+]/[AH] \qquad (2.8.1.d)$$

$$BOH \leftrightarrow B^+ + OH^- \qquad k_b = [B^+].[\,OH^-]/[BOH] \qquad (2.8.1.e)$$

F. Bazele ca acceptori de protoni

O substanță poate avea caracter bazic fără a elibera direct ioni OH în soluție apoasă. Este suficient a accepta protoni (ca opus al eliberării de protoni de către acizi) protonii acceptați provin dintr-o moleculă de apă, astfel încât în soluție, în mod indirect apare un ion OH^-. Ex.: amoniacul NH_3 are caracter alcalin:

$$NH_3 + H_2O \leftrightarrow NH_4^+ + OH^- \qquad (2.8.1.f)$$

2.8.2. Produsul ionic al apei

A. În cazul apei, în condiții normale (p = 1 atm, t = 20°C) produsul $[H^+].[OH^-]$ se numeşte produs ionic al apei şi are valoarea 10^{-14}.

$$H_2O \leftrightarrow H^+ + OH^- \qquad [H^+].[OH^-] = k_w\,[H_2O] = 10^{-14} \qquad (2.8.2.a)$$

B. Cum fiecare moleculă de apă eliberează în mod egal un ion de H^+ şi unul de OH^-, rezultă că:

$$[H^+] = [OH^-] = 10^{-7} \quad (2.8.2.b)$$

Cu alte cuvinte 1 moleculă din 10 milioane este disociată!

C. Ionul de hidroniu

De menționat că ionul H^+ nu se găsește niciodată liber în soluție, ci este întotdeauna cuplat pe o moleculă de apă formând ionul de hidroniu H_3O^+. Pentru simplitate în abordarea formală păstrăm însă notația de H^+.

2.8.3. Scara pH

A. Definiția pH-ului

Exprimarea concentrației ionilor H^+ într-o soluție se face prin definirea scării pH.

Definiție: pH-ul unei soluții este logaritmul cu semn schimbat al concentrației ionilor de hidrogen.

$$pH = - \log [H^+] \quad (2.8.2.c)$$

Se aplică logaritmi zecimali. În cazul apei cunoscând $[H^+]= 10^{-7}$ rezultă pH = 7. Pentru un acid tare cu concentrația 1 mol/l, pH = 0.

B. Definiția pOH-ului

Similar, logaritmul cu semn schimbat al concentrației ionilor OH^- se numește pOH.

$$pOH - \log OH \quad (2.8.2.d)$$

C. Relația pH – pOH

Prin logaritmarea formulei 2.8.2.a se obține relația:

$$pH + pOH = 14 \quad (2.8.2.e)$$

Din acest motiv, nu este necesar a lucra cu ambele mărimi, fiind suficientă una. În practică se folosește doar scara pH.

D. Descrierea scării pH

Caracterul acid, neutru sau alcalin al unei soluții se poate recunoaște după valoarea pH-ului. Scara pH, stabilită de Sorensen are valori între 0 și 14 și este ilustrată în figura 2.8.3, în care sunt redate și câteva exemple de valori de pH ale unor soluții mai des întâlnite.

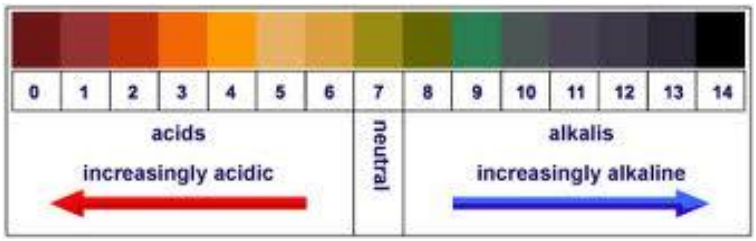


Fig. 2.8.3. Scara pH

E. Importanța pH-ului în materia vie

Există substanțe care au caracter amfoter, adică disociază atât ca acid cât și ca bază. În această categorie intră și amino - acizii. Gradul de disociere al grupărilor carboxil sau

amino va depinde de pH-ul soluției. Însă gradul de disociere determină şi sarcina electrică a moleculei, deci o serie de proprietăți. De aceea, organismele vii au mecanisme de reglare pentru menținerea pH-ului la valori relativ constante.

F. Soluții tampon

Soluțiile tampon sunt soluții care mențin constantă valoarea pH-ului. Ele sunt de regulă compuse dintr-un acid slab şi o sare a sa, sau dintr-o bază slabă şi o sare a sa. Să luăm primul caz:

$$AH \leftrightarrow A^- + H^+ \qquad (2.8.2.f)$$
$$AMe \leftrightarrow A^- + Me^+$$

Când în soluție se adaugă un acid deci apare o abundență de ioni H^+, echilibrul primei reacții se mută la stânga, în soluție existând suficienți ioni A^- proveniți din sare. În consecință se formează noi molecule neutre AH şi nu creşte concentrația ionilor H^+.

În cazul în care se adaugă o bază, procesul este invers, ionii H^+ neutralizați de ionii OH^- din bază, generând molecule de apă, sunt refăcuți prin disocierea altor molecule de AH, echilibrul mutându-se la dreapta.

pH-ul unei soluții tampon se calculează cu relația Henderson-Hasselbach:

$$pH = pk_a + \log([\text{f.deprot.}]/[\text{f.protonata}]) \qquad (2.8.2.g)$$

unde [f. deprot.] reprezintă concentrația formei deprotonate, iar [f. protonata] reprezintă concentrația formei protonate.

Dacă luăm $pk_a = -\log k_a$, observăm că atunci când sistemul este în echilibru, $pH = pk_a$ iar concentrația formei protonate este egală cu cea a formei deprotonate.

În exemplul de mai sus, forma protonată este AH iar forma deprotonată este A^- sau AMe (sărurile disociază total).

2.9. Termodinamica biologică

Unul dintre capitolele importante din biofizica celulară este cel referitor la termodinamica biologică. Vom urmări în continuare câteva deosebiri esențiale între materia nevie şi materia vie din punct de vedere termodinamic. Să revedem însă câteva noțiuni introductive de termodinamică.

2.9.1. Parametrii de stare ai unui sistem termodinamic

A. Obiectul de studiu în termodinamică se numeşte sistem termodinamic. El este definit ca o porțiune finită din univers, delimitată fizic sau imaginar de restul universului, numit mediu exterior.

B. Un sistem termodinamic este caracterizat la un moment dat printr-un set de parametri numiți parametri de stare. De ex.: un gaz într-un cilindru cu piston este un sistem termodinamic, caracterizat prin parametrii de stare presiune p, temperatură T şi volum V.

C. Parametrii de stare pot fi:

– *extensivi* – care depind de dimensiunea sistemului, de ex.: volumul, masa, numărul de moli, energia internă, etc.,

– *intensivi* – care nu depind de dimensiunea (masa) sistemului de ex.: temperatura, presiunea, densitatea, etc.

Parametrii extensivi au proprietatea de aditivitate, cei intensivi nu. Dacă asociem două sisteme 1 şi 2 pentru a crea un nou sistem, 3, vom avea masa finală $m_3 = m_1 + m_2$ energia internă $U_3 = U_1 + U_2$ ş.a.m.d.

D. Sistem omogen, gradienţi, sistem izolat

Când parametrii intensivi au aceeaşi valoare în orice punct din sistem spunem că sistemul este omogen. Când sistemul nu este omogen, putem defini variaţia unui parametru intensiv Y de-a lungul unei axe x prin gradientul parametrului respectiv dY/dx.

E. Un sistem termodinamic care nu poate schimba cu exteriorul nici substanţă, nici energie se numeşte sistem izolat. În cazul în care sistemul nu schimbă substanţă, dar poate schimba energie sub formă de lucru mecanic sau căldură sistemul se numeşte închis. În contrast cu sistemul închis este sistemul deschis, care permite şi schimb de substanţă. Un caz aparte de sistem închis este sistemul izolat adiabatic, care nu schimbă substanţă, iar energia poate fi schimbată numai sub formă de lucru mecanic, nu şi sub formă de căldură.

2.9.2. Procese termodinamice

A. Stare de echilibru. Starea unui sistem în care parametrii de stare rămân constanţi în timp şi prin sistem nu circulă fluxuri se numeşte stare de echilibru.

B. Stare staţionară - este starea unui sistem care are parametrii de stare constanţi în timp, dar sistemul este traversat de fluxuri constante.

C. Trecerea unui sistem dintr-o stare în altă stare se numeşte transformare sau proces termodinamic.

D. Un proces este caracterizat prin mărimi de proces, numite variaţii. Dacă (1) şi (2) sunt două stări distincte ale sistemului în două momente, t_1 şi t_2, atunci variaţia unui parametru de stare X va fi $\Delta X = X_2 - X_1$, care pentru intervale scurte de timp devine dX/dt. Variaţiile pot fi definite atât pentru parametrii intensivi cât şi pentru cei extensivi.

E. Există procese în care unii parametri rămân constanţi; iată denumirile câtorva procese particulare:
– transformarea izotermă – când temperatura rămâne constantă,
– transformarea izobară – când presiunea rămâne constantă,
– transformarea izocoră – când volumul rămâne constant.

F. Procese reversibile şi ireversibile

Un proces termodinamic între două stări 1şi 2 se numeşte reversibil dacă sistemul poate reveni din starea 2 în starea 1 prin aceleaşi stări intermediare. În caz contrar este proces ireversibil. Procesele reversibile sunt ideale, procesele reale sunt ireversibile, dar putem avea procese care se apropie destul de mult de procesele reversibile.

2.9.3. Funcţii de stare

Procesele termodinamice sunt mai uşor descrise cu ajutorul unor anumite funcţii de stare, pe care le enumerăm fără a le defini sau analiza proprietăţile:

- entropia termodinamică S (exprimată în J/K)
- energia internă U (exprimată în J)
- energia liberă F = U – T.S (J)
- entalpia H = U + p.V (J)
- entalpia liberă G = U + p.V – T.S (J).

2.9.4. Principiul al doilea al termodinamicii

Cel mai controversat aspect al termodinamicii biologice este legat de al doilea principiu al termodinamicii. Vom face câteva scurte comentarii legate de acest principiu.

A. Enunțul său din fizica clasică are mai multe formulări echivalente.

a) Căldura nu poate trece de la sine de la un corp cu temperatură mai scăzută la unul cu temperatură mai ridicată

b) În procesele termodinamice entropia nu poate scădea; variația entropiei este zero în procese reversibile şi pozitivă în procesele ireversibile.

B. Entropia termodinamică poate fi corelată intuitiv (demonul lui Maxwell) cu gradul de ordine la scară moleculară. În procesele ireversibile creşte dezordinea moleculară (numărul de stări în care putem aranja moleculele componente) asociată cu creşterea entropiei. Acest lucru este valabil pentru sisteme nevii.

C. Însă procesele din materia vie sfidează (aparent) al doilea principiu al termodinamicii: structurile evoluează către stări tot mai ordonate!

Explicația poate fi dată prin legătura care se face între entropia termodinamică şi cea informațională. Un proces termodinamic însoțit de scăderea entropiei termodinamice ar fi posibil dacă sistemul îşi creşte entropia informațională.

2.10. Procese cuplate

2.10.1. Natura proceselor cuplate

A. Procesele din materia vie nu sunt procese izolate ci procese cuplate. Este deci posibil a avea cuplate două procese – unul în care entropia scade (reacții de sinteză) cuplat cu unul în care entropia creşte (un proces catabolic, spontan), respectând şi principiul al II – lea prin creşterea globală a entropiei în ansamblul proceselor cuplate.

B. Din punct de vedere energetic un proces care necesită energie (endoenergetic) trebuie cuplat cu un proces ce eliberează energie (exoenergetic).

2.10.2. Laturile metabolismului

Procesele cuplate le întâlnim ca laturi principale ale metabolismului.

A. *Catabolismul* – cuprinde procese exoenergetice, de degradare moleculară, în care entropia creşte sunt procese spontane.

B. *Anabolismul* – cuprinde latura specifică a materiei vii, procese endoenergetice în care se produc structuri mai ordonate şi mai bogate în energie în aceste procese entropia scade (reacții de sinteză, endoenergetice).

2.10.3. Stocarea energiei pentru procesele biologice

Pentru a asigura energia necesară reacţiilor anabolice, în materia vie energia este stocată în molecule de ATP (acid adenozin-trifosforic). Aceste molecule, numite şi molecule macroergice, sunt sintetizate în organitele celulare numite mitocondrii, printr-un proces numit fosforilare oxidativă. Descifrarea mecanismelor sintezei ATP prin pompa de protoni de către Mitchell a fost răsplătită cu un premiu Nobel, în 1974.

De fapt importanţa proceselor termodinamice, înţelegerea lor în contextul materiei vii, a generat o bună conturare a termodinamicii biologice, numită şi termodinamica proceselor ireversibile, pentru care s-a acordat premiul Nobel lui Prigogine, în 1971.

2.11. Forţe termodinamice

A. Forţe şi fluxuri termodinamice

În termodinamica biologică se generalizează termenul de forţă, numită forţa termodinamică, definită ca gradient al unui parametru intensiv. Efectul produs de o forţă termodinamică se numeşte flux conjugat şi se defineşte ca transferul unei mărimi print-o secţiune unitară, în unitatea de timp.

B. Lista forţelor termodinamice şi fluxurile lor conjugate

Tabelul 2.11. Forţe şi fluxuri termodinamice

Forţa	Param. intensiv	Flux conjugat	Flux
Δp	gradient de presiune	flux de volum (curgere)	J_V
ΔT	gradient de temperatură	flux de căldură	J_Q
ΔE	gradient de potenţial electric	curent electric	I
ΔC	gradient de concentraţie (fiecare subst. i)	difuziune	J_i
$\Delta \pi$	gradient de presiune osmotică	osmoză	J_w
$\Delta \mu$	gradient de potenţial chimic	rata reacţiei chimice	v

2.12. Transport transmembranar

2.12.1. Clasificare, proprietăţi

O serie de molecule importante în studiile de bioinformatică le întâlnim în structurile care asigură transportul substanţelor prin membranele biologice.

Transportul prin membranele biologice poate fi:
- transport pasiv în sensul gradientului electrochimic, fără consum energetic
- transport activ contra gradientului electrochimic, cu consum energetic.

2.12.2. Transport pasiv

Transportul pasiv se poate realiza prin:

a) difuziune – în cazul gazelor sau substanţelor solubile în lipide

b) prin canale ionice – există structuri moleculare de natură proteică, ce traversează membrana celulară, având forma unui canal, adesea având situri de cuplare temporară a ionilor. În figura 2.12.2 este prezentat un canal de Na^+.

c) transport facilitat – cu ajutorul unor molecule ”carrier” pentru molecule pentru care nu există nici canale, nici nu pot difuza, de ex. transportul glucozei prin membrana eritrocitară.

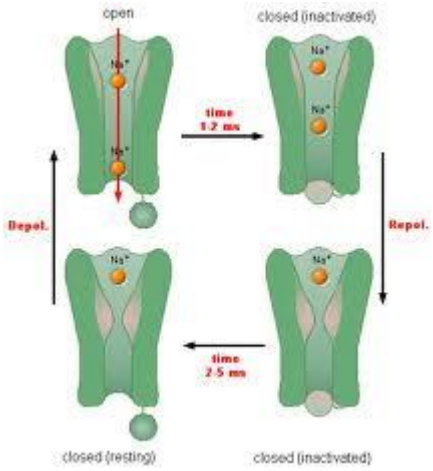


Fig. 2.12.2. Canal de Na^+

2.12.3. Transport activ

A. Transportul activ are un rol important în menţinerea parametrilor fizico-chimici ai celulelor. El este asigurat de structuri specializate numite pompe, fiind frecvente pompele pentru ioni.

Structura moleculelor ce formează pompele este mai complexă decât a canalelor, ele având şi situsuri de cuplare a moleculelor ce asigură suportul energetic. De obicei procesul începe cu o reacţie de fosforilare, din partea unei molecule de ATP.

B. În figura 2.12.3 este prezentată pompa Na^+/K^+, întâlnită cel mai frecvent în celule pentru a asigura concentraţiile normale intra şi extra celulare de Na^+ şi K^+.

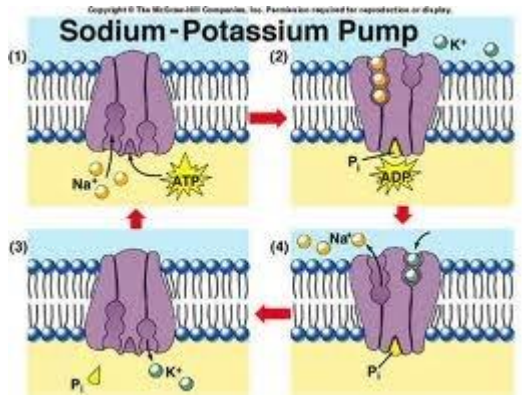


Fig. 2.12.3. Pompa Na^+/K^+

3. Noțiuni de biochimie

În acest capitol vom trece în revistă structura şi proprietățile a două mari clase de molecule ce constituie de fapt obiect de studiu pentru bioinformatică:

– proteinele, compuse din aminoacizi,
– acizi nucleici, compuşi din nucleotide.

3.1. Aminoacizii

3.1.1. Structură generală

A. Aminoacizii (AA) sunt molecule organice care conțin o grupare amino ($-NH_2$) şi o grupare acidă carboxil (–COOH).

B. Formula generală este H_2N-CH-R-COOH. Atomul de carbon primar (numit şi C_α) are cele 4 valențe ocupate astfel: una leagă gruparea amino, a doua leagă gruparea carboxil, a treia leagă un proton (- H) iar a patra leagă o grupare numită radical sau reziduu (-R). Aminoacizii diferă între ei prin acest radical (figura 3.1.1.a).

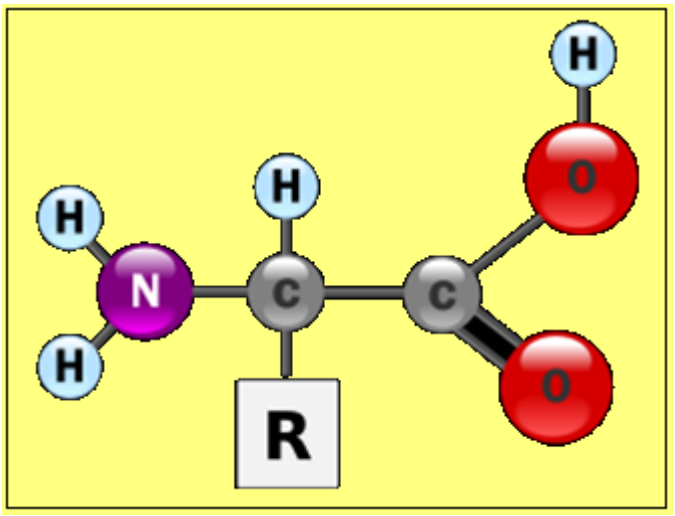


Fig. 3.1.1.a. Formula generală a aminoacizilor

C. În materia vie întâlnim doar 20 AA, aceiaşi atât în regnul vegetal cât şi animal, la toată scara evolutivă, demonstrând elocvent unitatea materiei vii pe pământ. Însă numărul practic infinit al combinațiilor posibile, acoperă cu prisosință orice paletă imaginativă, explicând toate diversitățile, la fel cum cu numărul restrâns al literelor alfabetului se pot crea orice cuvinte.

D. În figura 3.1.1.b sunt prezentate structurile moleculare ale aminoacizilor în forma lor uzuală.

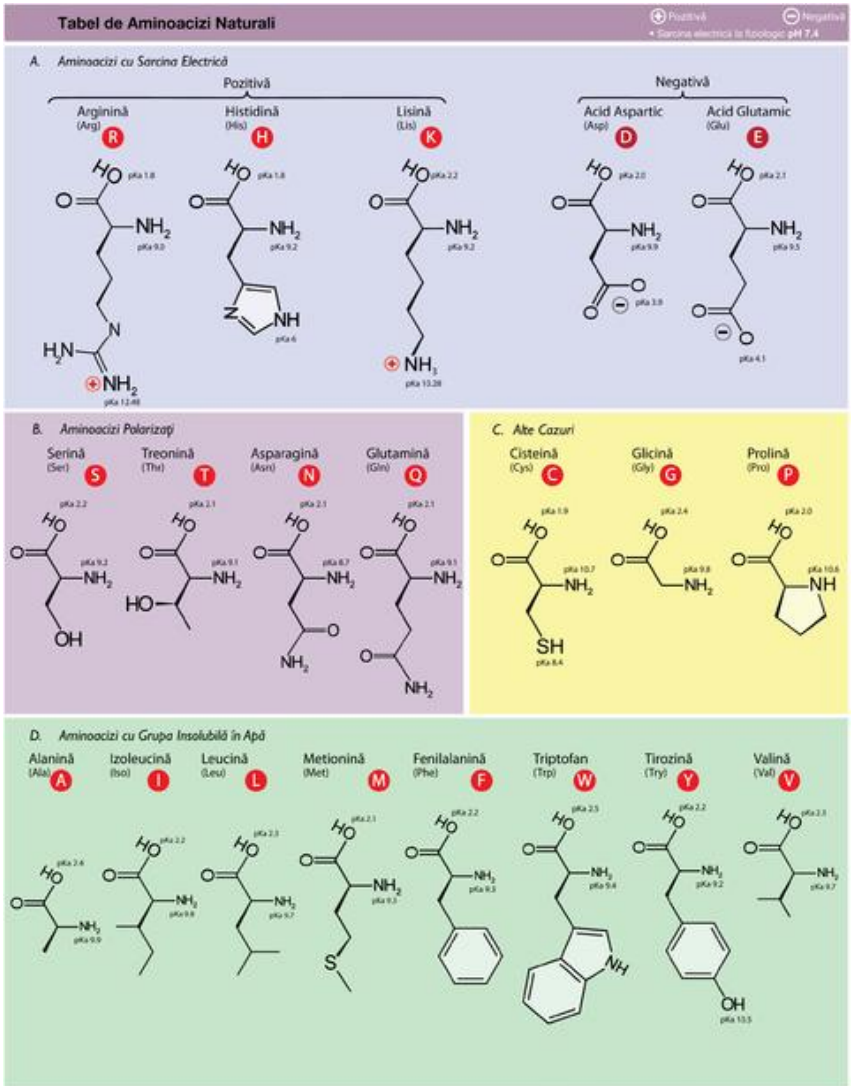


Fig. 3.1.1.b. Structurile moleculare ale aminoacizilor

3.1.2. Formele ionice ale aminoacizilor

A. Gruparea carboxil are caracter acid, putând elibera un proton şi trecând în forma $–COO^-$, cu sarcină negativă.

B. Gruparea amino are ca caracter bazic, putând accepta un proton şi trecând în forma $–NH_3^+$, cu sarcină pozitivă.

C. Având caracter dublu, atât acid cât şi alcalin, spunem că are un caracter amfoter.

D. Formele ionice care se pot forma, fiind atât pozitive cât şi negative, cu posibilitatea ca o moleculă să prezinte simultan ambele sarcini, poartă denumirea de "zwitter-ioni", şi sunt prezentate în figura 3.1.2.

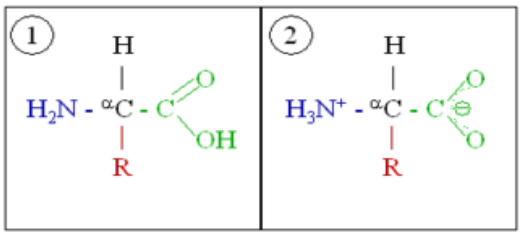


Fig. 3.1.2. Forme ionice ale aminoacizilor

E. Sarcina pe care o are un aminoacid la un moment dat depinde de pH-ul mediului. În mediu acid, unde avem o abundenţă de protoni H^+, gruparea carboxil nu disociază, în schimb gruparea amino se va găsi în formă ionică, iar aminoacidul va avea sarcină pozitivă. În schimb în mediu bazic, gruparea amino va fi neutră, iar cea carboxil ionizată, având o sarcină globală negativă.

F. Pentru fiecare aminoacid există o anumită valoare a pH-ului, numită *punct izoelectric* pentru care gradul de disociere ca acid este egal cu cel de disociere ca bază, numărul sarcinilor pozitive este egal cu cel al sarcinilor negative, iar molecula în ansamblu este neutră.

G. Dependenţa sarcinii de pH stă la baza unei metode de separare a proteinelor numită *electroforeză*. O bandă de hârtie de filtru îmbibată în soluţie salină are marginile în contact cu doi electrozi între care se aplică o diferenţă de potenţial de ordinul sutelor de volţi. Pe bandă se depune o picătură din soluţia unei proteine. În prezenţa câmpului electric moleculele se deplasează (migrează) în funcţie de sarcina lor. Viteza de deplasare este invers proporţională cu masa moleculară. Această deplasare datorată câmpului electric se numeşte electroforeză.

3.1.3. Proprietăţile aminoacizilor

A. Abundenţă

Cei 20 AA nu au o răspândire uniformă în proteinele din natură. Abundenţa lor (în procente) este prezentată în tabelul 3.1.3.

B. Izomerie structurală

Dacă se trasează un plan imaginar creat de carbonul α, carbonul din carboxil şi azotul din amino, atunci pentru fiecare AA avem două posibilităţi de poziţionare a radicalului R - fie de partea stângă, fie de partea dreaptă a planului, cu alte cuvinte avem 2 izomeri sterici, care poartă numele S şi R, iar proprietatea de a prezenta izomeri sterici se numeşte chiralitate.

Aminoacizii din proteinele din natură sunt cu toţii izomeri S, acceptând cisteina (Cys) care este R şi glicina (Gly) care este nonchirală.

C. Izomerie optică

Compuşii care prezintă izomeri structurali, prezintă de obicei şi izomerie optică. O serie de substanţe (numite „optic active") au proprietatea de a roti planul luminii polarizate când această lumină le traversează. În funcţie de sensul în care este rotit planul luminii polarizate substanţele pot fi D (dextrogire) - care rotesc planul la dreapta şi L (levogire) care rotesc planul la stânga. În aminoacizii de sinteză cele două forme se produc în proporţii egale. În aminoacizii naturali întâlnim numai forma L. Fenomenul încă nu are o explicaţie unanim acceptată.

Tabel 3.1.3. Răspândirea aminoacizilor

Denumirea (Residue)	cod 3-litere	cod 1 literă code	Abundență />(%) E.C.
Alanină	ALA	A	13.0
Arginină	ARG	R	5.3
Asparagină	ASN	N	9.9
Aspartat	ASP	D	9.9
Cisteină	CYS	C	1.8
Acid glutamic	GLU	E	10.8
Glutamină	GLN	Q	10.8
Glicină	GLY	G	7.8
Histidină	HIS	H	0.7
Isoleucină	ILE	I	4.4
Leucină	LEU	L	7.8
Lizină	LYS	K	7.0
Metionină	MET	M	3.8
Fenilalanină	PHE	F	3.3
Prolină	PRO	P	4.6
Serină	SER	S	6.0
Treonină	THR	T	4.6
Triptofan	TRP	W	1.0
Tirosină	TYR	Y	2.2
Valină	VAL	V	6.0

D. Aminoacizii esențiali

Majoritatea aminoacizilor pot fi sintetizați în organismul uman. Totuşi există unii AA care nu pot fi sintetizați – aceştia poartă numele de AA esențiali. Ei sunt în număr de 9: Val, Leu, Ile, Lys, Thr, Met, His, Trp, Phe. AA sunt sintetizați de plante şi sunt necesari în aportul alimentar, ei se găsesc şi în produse animale.

E. Notații

Se folosesc frecvent nişte notații prescurtate, formate din 3 litere convenabile pentru recunoaşterea denumirii lor. Totuşi în analiza secvențială este preferată notația simbolică cu câte 1 literă.

3.1.4. Hidrofobicitatea

A. Grupările din radicali R sunt foarte variate, cu proprietăți diferite, ceea ce conferă aminoacizilor – şi prin ei, proteinelor – o paletă largă de proprietăți. În funcție de aceste grupări putem face o clasificare a AA (figura 3.1.4.a)

- AA cu grupări nepolare: Gly, Ala, Val, Leu, Ile, Phe,
- AA cu grupări polare neîncărcate electric: Ser, Thr, Cys, Met, Asn, Gln, Tyr, Trp, Pro,
- AA cu grupări polare cu caracter acid: Asp, Glu,
- AA cu grupări polare cu caracter bazic: Arg, Lys.

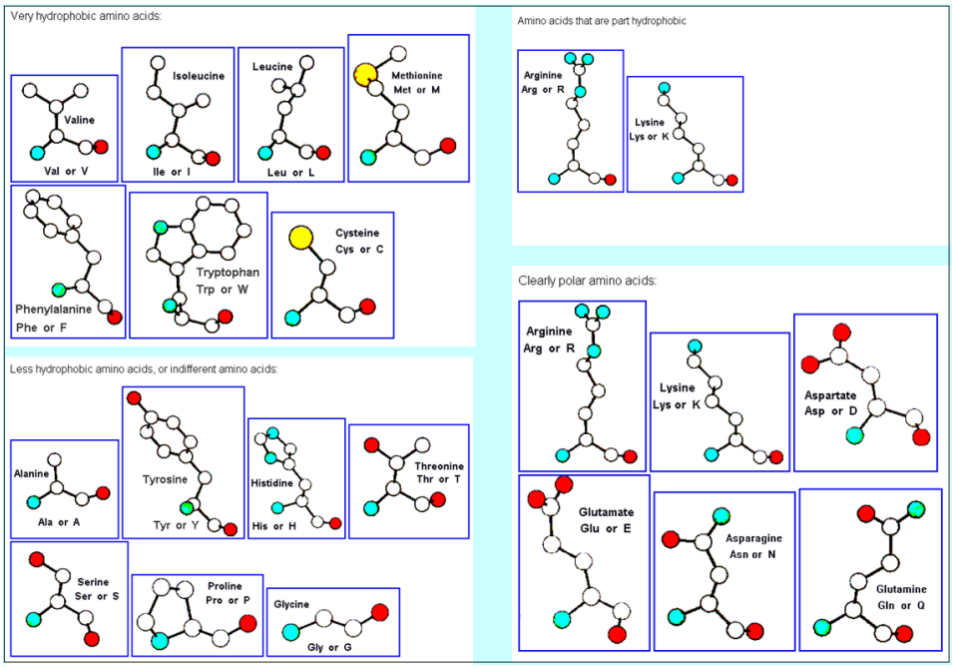


Fig. 3.1.4.a. Clasificarea aminoacizilor

B. Polaritatea grupărilor determină şi solubilitatea lor în apă. Astfel AA cu grupări polare sunt hidrofobi, cei cu sarcină sunt hidrofili, iar cei cu grupări polare neîncărcate electric sunt relativ solubili prin punțile de hidrogen care se pot forma.

C. Scări de hidrofobicitate

Evaluarea gradului de hidrofobicitate al unei molecule este dificilă, fiind cel mai adesea bazată pe partiția moleculei între diferiți solvenți. În funcție de solvenții folosiți s-au stabilit diverse scări care ierarhizează aminoacizii figura 3.1.4.b, dreapta.

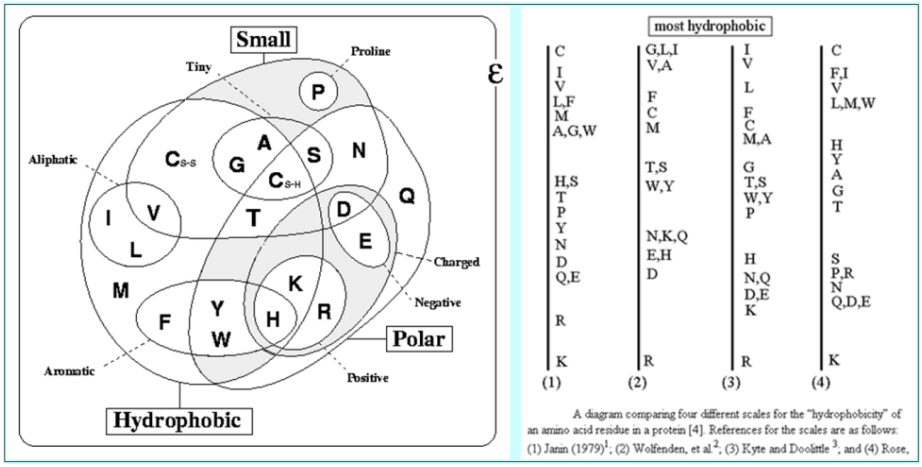


Fig. 3.1.4.b. Hidrofobicitatea aminoacizilor

D. Dacă ţinem cont de dimensiunea moleculei, pe lângă hidrofobicitate şi caracterul polar/nepolar sau sarcina moleculei la pH fiziologic, putem prezenta schematic o diagramă care sintetizează toate aceste proprietăţi figura 3.1.4.b stânga.

E. Pentru a avea o imagine mai detaliată asupra proprietăţilor aminoacizilor, să mai adăugăm determinările unor proprietăţi fizice:densitate, volum, suprafaţă, constante de echilibru de disociere a grupărilor din radical (unde este cazul). Aceste proprietăţi sunt sintetizate în tabelul 3.1.4.

Tabelul 3.1.4. Proprietăţile aminoacizilor solubilitate, densitate, volum, suprafaţă şi pK

Name	Solubility (g/100g, 25 °C)	Crystal Density (g/ml)	pI at 25 °C
Alanine	16.65	1.401	6.107
Arginine	15	1.1	10.76
Aspartic Acid	0.778	1.66	2.98
Asparagine	3.53	1.54	-
Cysteine	very	-	5.02
Glutamic Acid	0.864	1.460	3.08
Glutamine	2.5	-	-
Glycine	24.99	1.607	6.064
Histidine	4.19	-	7.64
Isoleucine	4.117	-	6.038
Leucine	2.426	1.191	6.036
Lysine	very	-	9.47
Methionine	3.381	1.340	5.74
Phenylalanine	2.965	-	5.91
Proline	162.3	-	6.3
Serine	5.023	1.537	5.68
Threonine	very	-	-
Tryptophan	1.136	-	5.88
Tyrosine	0.0453	1.456	5.63
Valine	8.85	1.230	6.002

	Residue Volume	Surface Area	Side Chain pKa
A	88.6	115	-
R	173.4	225	~12
D	111.1	150	4.5
N	114.1	160	-
C	108.5	135	9.1-9.5
E	138.4	190	4.6
Q	143.8	180	-
G	60.1	75	-
H	153.2	195	6.2
I	166.7	175	-
L	166.7	170	-
K	168.6	200	10.4
M	162.9	185	-
F	189.9	210	-
P	112.7	145	-
S	89.0	115	-
T	116.1	140	-
W	227.8	255	-
Y	193.6	230	9.7
V	140.0	155	-

a-amino pKa = 6.8 - 7.9, a-carboxyl pKa = 3.5 - 4.3

3.2. Proteine

3.2.1. Legătura peptidică

A. Doi aminoacizi se pot lega chimic prin reacţia între gruparea carboxil a unui AA cu gruparea amino a celuilalt, cu eliminarea unei molecule de apă:

$$H_2N\text{-}CH\text{-}R_1\text{–}COOH + H_2N\text{-}CH\text{-}R_2\text{–}COOH \rightarrow H_2N\text{-}CH\text{-}R_1\text{–}CO\text{-}NH\text{-}CH\text{-}R_2\text{–}COOH \qquad (3.2.1.a)$$

B. Legătura formată se numeşte legătură peptidică. Reacţia este bine ilustrată în figura 3.2.1.a, iar proprietăţile legăturii peptidice sunt prezentate în figura 3.2.1.b.

C. Ne putem uşor imagina că dipeptidul format, având un capăt amino şi un capăt carboxil, poate la rândul său forma o nouă legătură peptidică şi aşa mai departe. O astfel

de moleculă se numeşte polipeptid sau proteină. Pentru un număr mic de AA în structură se pot folosi prefixele corespunzătoare: dipeptid, tri~, tetra~, penta~ ş.a.m.d.

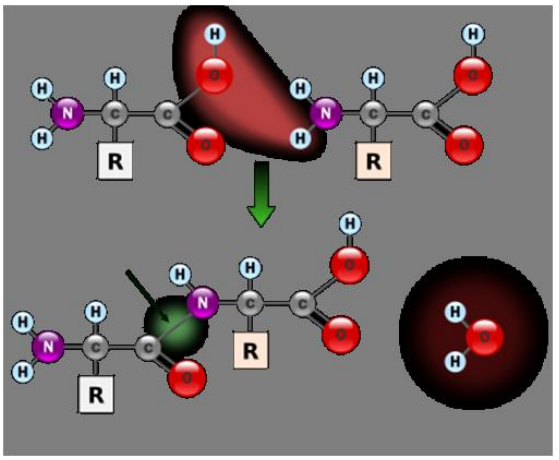


Fig. 3.2.1.a. Formarea legăturii peptidice

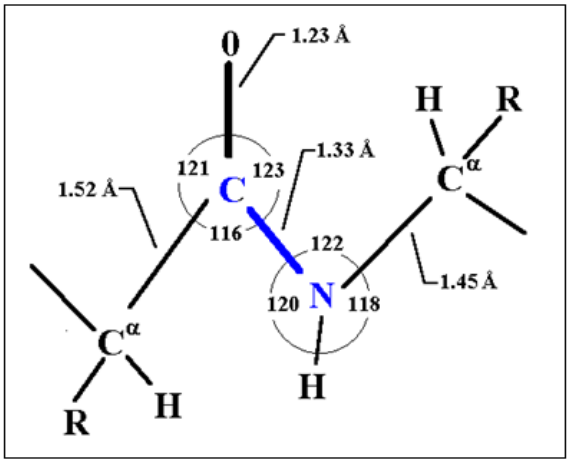


Fig. 3.2.1.b. Proprietățile legăturii peptidice

3.2.2. Structura primară a proteinelor

A. O primă caracteristică, de importanță majoră în bioinformatică este reprezentată de succesiunea de AA care formează molecula proteică.

B. Această succesiune poartă numele de „structură primară" a proteinelor. Pozițiile aminoacizilor se numerotează începând de la capătul cu gruparea amino liberă.

C. Determinarea structurii primare a proteinelor

Există mai multe metode de determinare a structurii primare a proteinelor, pe care nu le vom trata în detaliu ci doar le vom enumera aici:

a) metode biochimice de determinare a secvenței, cuprinzând:
- degradarea Edman,
- fracționarea prin digestie cu enzime proteolitice,

- spectrometrie de masă,
- hidroliză în acizi (doar determinarea cantităților de aminoacizi).

b) deducerea din secvența genelor.

D. Exemple de structuri primare

Una dintre primele structuri primare determinate a fost cea a insulinei, prezentată în figura 3.2.2, de către Sanger în 1951, pentru care i s-a acordat premiul Nobel.

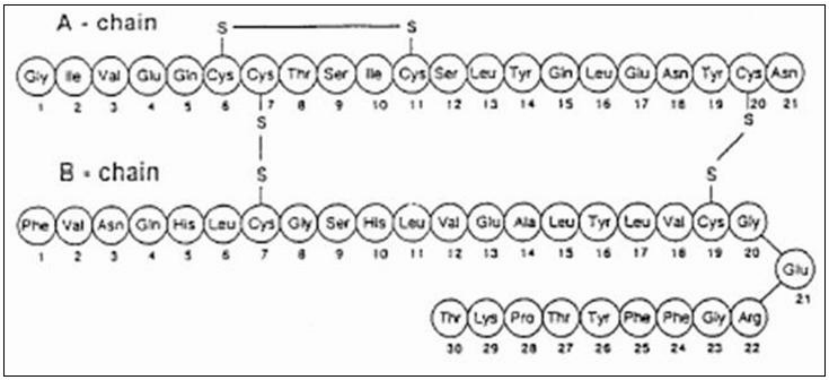


Fig. 3.2.2. Structura primară a proteinelor: insulina

3.2.3. Structura secundară a proteinelor

A. O moleculă proteică poate avea de la câteva zeci de aminoacizi până la câteva sute. Prin proprietățile pe care le au radicalii aminoacizilor din lanț, se formează adesea niște structuri spațiale specifice uşor de recunoscut. Corey şi Pauling (1943) au fost primii care au semnalat aceste structuri denumite generic „structura secundară" a proteinelor.

B. Structura – helix

Cea mai bine cunoscută formă specifică este cea de helix pe care o întâlnim frecvent la proteine. Unii aminoacizi, mai ales Met, Ala, Leu, Glu şi Lys se dispun în forma unor spirale ce s-ar înfăşura pe un cilindru virtual, numit - helix. Alți AA însă nu se dispun în helix, apariția lor (de exemplu Gly şi Pro) generând discontinuitatea spiralei, de aceea se şi numesc "helix breakers". Procentul de AA angajați în structurile secundare variază. În tabelul 3.2.3. sunt trecute proporțiile de AA cuprinşi în structuri de tip - helix.

Tabelul 3.2.3. Procent de AA în elice

Proteina	**% elice α**
Mioglobina	70
Insulina	38
Ovalbumina	31
Serumalbumina	46
Pepsina	31
Ribonucleaza	16
Chimotripsina	15

C. Să trecem în revistă câteva proprietăți ale unui α – helix (figura 3.2.3.a)

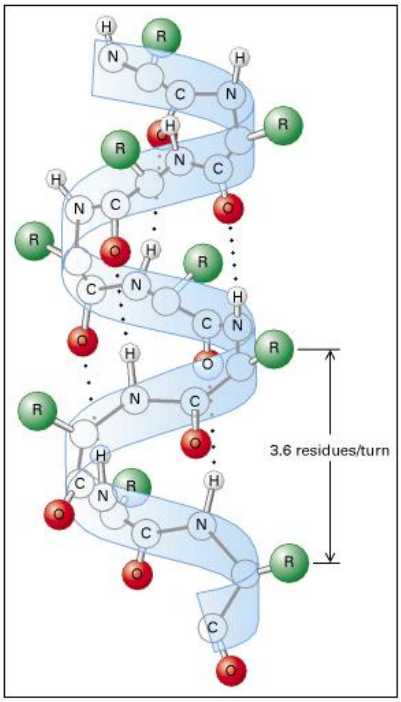


Fig. 3.2.3.a. Structura α helix a proteinelor

- o spiră cuprinde 3,6 AA,
- pasul spirei 5,21 Å,
- forma spiralei ca un şurub cu pasul spre dreapta,
- catenele (radicalii R) sunt orientate spre exteriorul cilindrului.

D. Determinarea structurii secundare

Cea mai potrivită metodă este difracția cu raze X, (utilizată de Perutz şi Kendrew pentru structura mioglobinei), radiațiile X având lungimea de undă de ordinul de mărime al distanțelor interatomice (~Å). O limitare a metodei este că poate fi folosită numai pe cristale, deci moleculele care nu pot fi aduse în formă cristalină nu pot fi studiate prin această metodă.

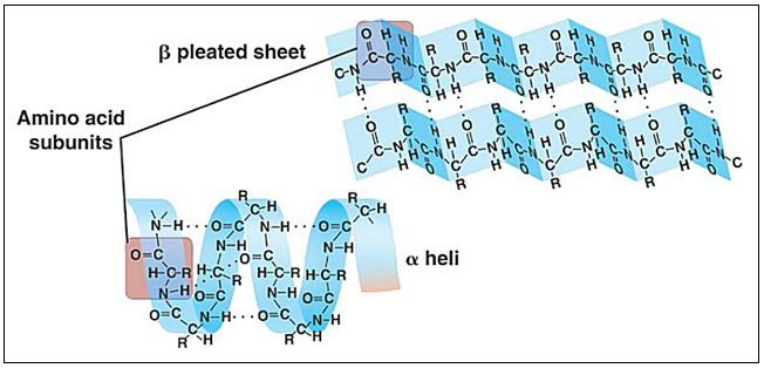


Fig. 3.2.3.b. Structura pliuri β a proteinelor

E. Fâşiile β

O altă formă întâlnită ca structură secundară este reprezentată de „fâşiile ß" sau „pliurile ß" (figura 3.2.3.b). Alternanța unor AA care generează alternanța orientării valențelor duce la formarea unor regiuni cu formă specifică de panglică pliată. Aceste fâşii apar în special în regiunile cu AA aromatici (Trp, Tyr şi Phe). De asemenea, forma ß-C este generată în zonele cu Ile, Val, Thr.

3.2.4. Structura terțiară a proteinelor

A. Pe lângă structura secundară, între diferitele părți ale unei molecule proteice pot apărea interacțiuni:

- legături între părți prin punți de hidrogen, legături disulfidice sau legături Van der Waals
- plieri, încovoieri.

Toate acestea generează o structură tridimensională complexă.

3.2.5. Structura cuaternară

A. În cazul moleculelor foarte mari se pot realiza chiar legături între mai multe unități, formând aşa numita structură cuaternară. Procesul acesta este catalizat de enzime specifice numite „ holoenzime".

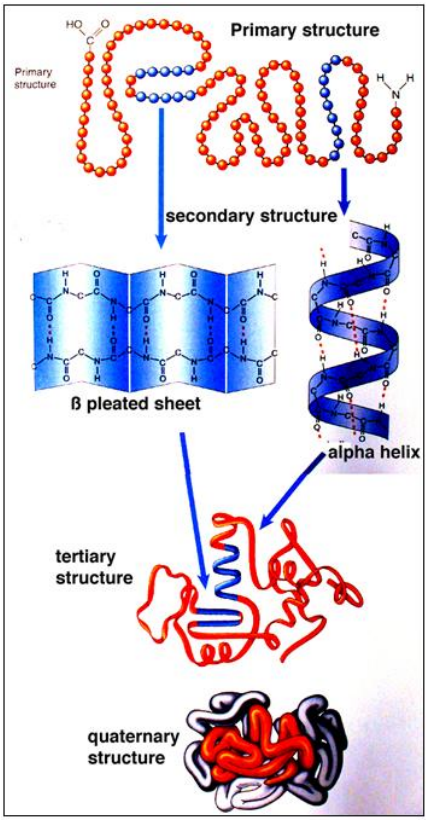


Fig. 3.2.5. Structura proteinelor – viziune de ansamblu

B. Exemple

Structură cuaternară întâlnim în special la proteinele globulare. Un exemplu: hemoglobina, care are 4 subunități, „molecula" fiind de fapt un tetramer.

În aceeaşi clasă mai putem include ADN polimeraza precum şi proteinele componente ale canalelor ionice.

În figura 3.2.5 este prezentată o sinteză privind structura proteinelor.

3.3. Componentele acizilor nucleici

3.3.1. Componentele unui nucleotid

Acizii nucleici au şi ei o structură secvențială, asemănătoare întrucâtva cu cea a proteinelor. Unitatea de secvență, echivalentă aminoacizilor, se numeşte „nucleotid". Un nucleotid are 3 componente principale:

- o pentoză,
- un acid fosforic,
- o bază azotată.

3.3.2. Pentozele din acizii nucleici

A. În acizii nucleici întâlnim două tipuri de pentoze:

- riboza - în acidul ribonucleic (ARN),
- de(z)oxiriboza - în acidul de(z)oxiribonucleic ADN.

B. Structura chimică şi numerotarea atomilor sunt prezentate în figura (3.2.2).

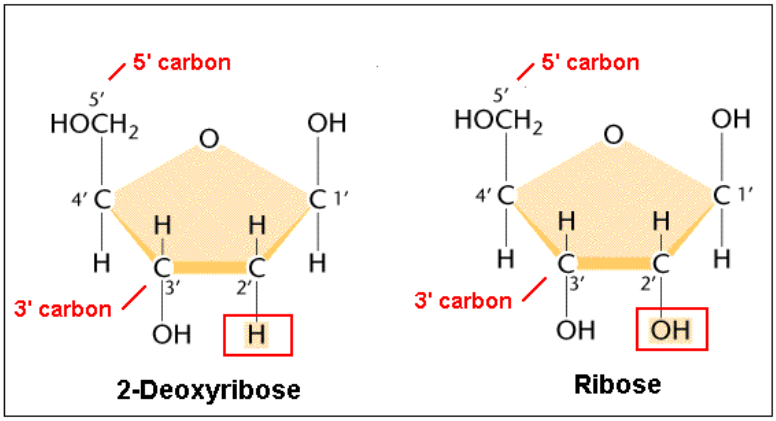


Fig. 3.2.2. Structura ribozei şi dezoxiribozei

Observăm că atomii C_1' - C_4' şi oxigenul formează un plan, în timp ce carbonul C_5' este înafara planului.

De asemenea, gruparea -OH de la carbonul 1' este de aceeaşi parte cu carbonul 5', în timp ce grupările -OH din pozițiile 2' şi 3' sunt de cealaltă parte a planului.

C. În moleculele de dezoxiriboză lipseşte gruparea -OH din poziția 2'.

3.3.3. Bazele azotate

A. Fiecare tip de acid nucleic conține 4 tipuri de baze azotate: două baze purinice şi două baze pirimidinice.

Bazele purinice sunt adenina (A) şi guanina (G) , iar cele pirimidinice sunt citozina (C) şi Timina (T) sau uracilul (U). În timp ce bazele A, G şi C se găsesc şi în ADN şi în ARN, cea de a patra este diferită: în ADN întâlnim timina, în timp ce în ARN întâlnim uracilul.

B. Numerotarea atomilor din bazele azotate se face numai pentru atomii din cicluri şi este redată împreună cu structura moleculelor de purină şi pirimidină, precum şi bazele azotate derivate: A şi G, respectiv C, T şi U în figura 3.3.3.

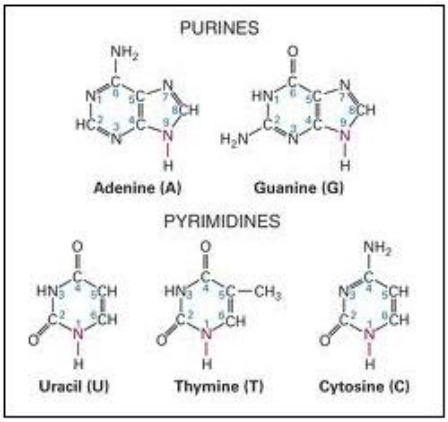


Fig. 3.3.3. Bazele azotate din acizii nucleici

3.3.4. Nucleozide

A. Bazele azotate se leagă de pentoze printr-o legătură glicozidică, formând o structură numită „nucleozid". Denumirea nucleozidelor porneşte de la numele bazei azotate; de ex.: de la adenină se formează adenozina (prin cuplarea de riboză) sau dezoxiadenozina (prin legarea de dezoxiriboză); similar avem guanină/guanozină, dezoxiguanozină, citozină/citidină/dezoxicitidină, timină/timidină (cu dezoxiriboza), respectiv uracil/uridină (cu riboza).

Legarea bazelor azotate se face întotdeauna la carbonul 1 al pentozei. Bazele purinice se leagă prin azotul din poziția 9, iar cele pirimidinice prin azotul din poziția 1.

B. Structura nucleozidelor este prezentată în figura 3.3.4.

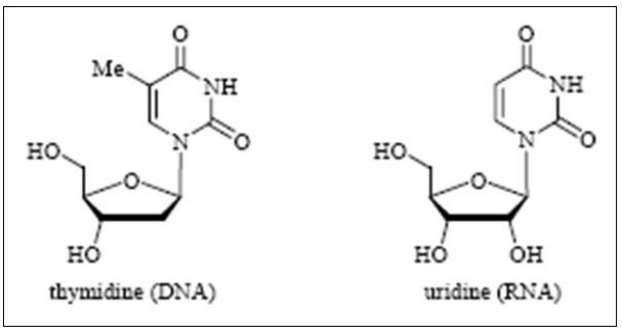


Fig. 3.3.4. Structura nucleozidelor

3.3.5. Nucleotide

Pentru formarea lanțurilor acizilor nucleici, nucleozidele sunt fosforilate. Cuplarea moleculei de acid fosforic se face la carbonul 5' al pentozei, structura nou formată fiind numită „nucleotid". Deşi nucleotidele, ca entități separate au denumiri specifice (acid adenilic, guanilic, citidilic sau timidilic, uneori sub formă de radical adenilat, guanilat, citidilat, timidilat) în cazul precizării unui nucleotid într-o secvență de acid nucleic, se foloseşte doar numele bazei azotate din componența sa. În figura 3.3.5 este redată structura nucleotidelor.

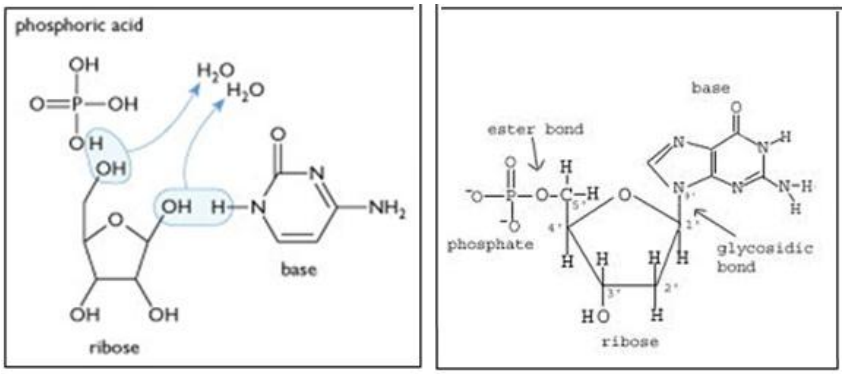


Fig. 3.3.5. Structura nucleotidelor

3.4. Structura acizilor nucleici

3.4.1. Formarea lanțului polinucleotidic

A. Nucleotidele se pot lega unele de altele prin eliminarea unei molecule de apă între o grupare - OH a fosforului legat de carbonul 5' şi gruparea OH a carbonului 3' din pentoză (figura 3.4.1.a).

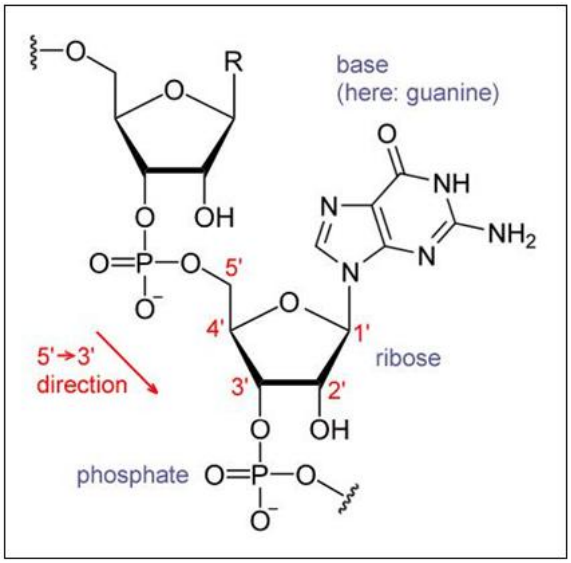


Fig. 3.4.1.a. Formarea unui dinucleotid

B. Prin repetarea procesului se poate obține un lanț care să cuprindă de la câteva zeci până la câteva mii de nucleotide (figura 3.4.1.b).

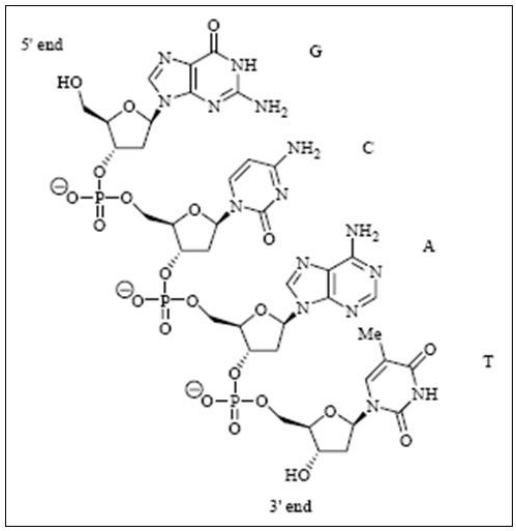


Fig. 3.4.1.b. Formarea lanțului polinucleotidic GCAT

3.4.2. *Structura primară a acizilor nucleici*

A. Toate lanțurile de acizi nucleici au două capete libere, unul la atomul 5', altul la atomul 3'. O secvență de lanț al unui acid nucleic se citește în ordinea 5' → 3'

B. Secvența bazelor azotate într-un lanț de acid nucleic poartă denumirea de structură primară a acizilor nucleici (figura 3.4.2).

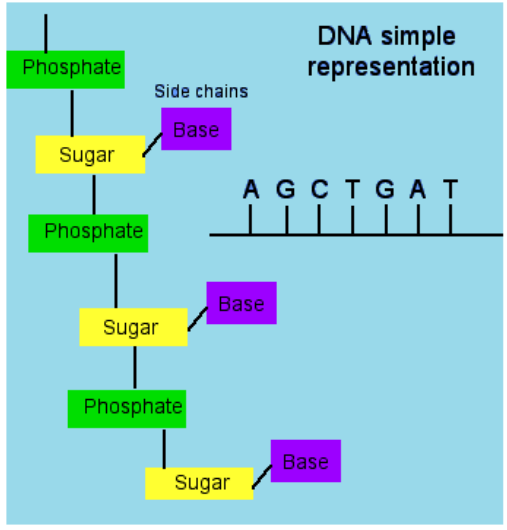


Fig. 3.4.2. Structura primară a unei secvențe ADN

3.4.3. Structura secundară a acizilor nucleici

A. Descifrarea structurii acizilor nucleici a reprezentat un pas important în înțelegerea unui întreg şir de fenomene din biologie în general şi din genetică în special. Acumularea unor date experimentale privind constanța rapoartelor A/T şi G/C, sau considerente teoretice privind structura tridimensională a moleculelor, au condus în cele din urmă la stabilirea modelului „dublu-helix" al moleculei de ADN de către Watson şi Crick în 1953, laureați ai premiului Nobel.

B. Moleculele de ADN formează o dublă elice înfăşurată pe un cilindru virtual, având orientate spre interior bazele azotate. Între două baze azotate din cele două lanțuri se stabilesc nişte punți de hidrogen. Prin configurația spațială a norului electronic sunt posibile doar perechi între o adenină de pe un lanț cu o timină pe lanțul opus (prin 2 punți de hidrogen) sau între o guanină şi o citozină (prin 3 punți de hidrogen) , aşa cum se poate vedea în figura 3.4.3.a.

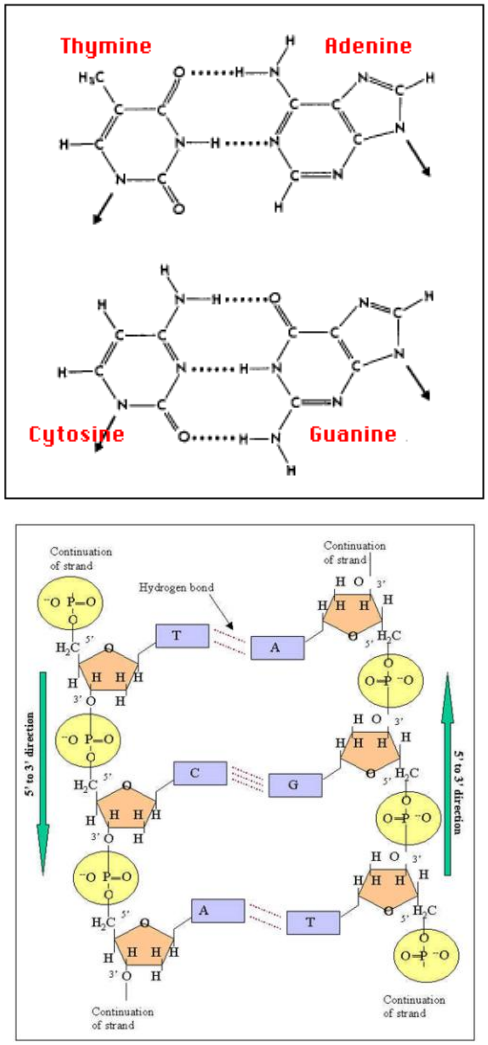


Fig. 3.4.3.a. Perechile de baze azotate în ADN

C. Proprietățile geometrice ale dublului helix au fost ulterior confirmate experimental și sunt redate sintetic în figura 3.4.3.b.

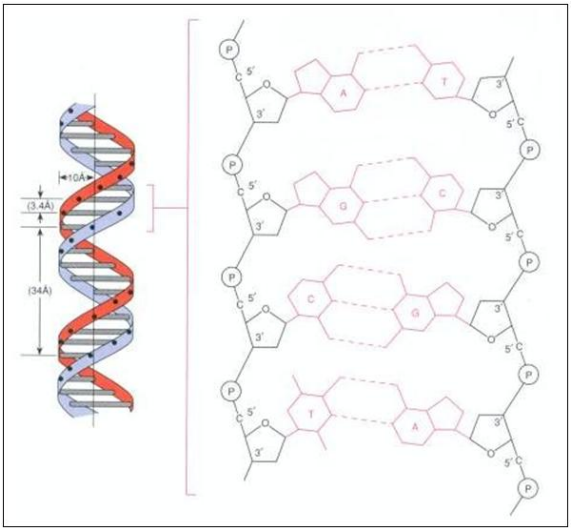


Fig. 3.4.3.b. Dublul helix ADN

D. Prezența legăturilor de hidrogen conferă o bună stabilitate moleculelor de acizi nucleici.

În moleculele de ARN, în locul timinei găsim uracilul.

4. Elemente de biologie celulară şi moleculară

O caracteristică esențială a materiei vii este reprezentată de structura sa celulară, alături de încă două proprietăți fundamentale: metabolismul şi reproducerea.

Deşi există mari diferențe între celulele diverselor specii, sau chiar între celulele aceluiaşi organism, există şi o serie de caracteristici comune pe care le vom trece în revistă.

Disciplina care se ocupă cu studiul celulei şi a fenomenelor la nivelul celular se numeşte „biologie celulară”. Din biologia celulară vom extrage sintetic câteva noțiuni privind:

- structura celulei umane (membrană, citoplasmă, nucleu, organite celulare – mitocondria şi ribozomii),
- diviziunea celulară mitoza şi meioza.

Explicarea fenomenelor biologice pe baze moleculare face obiectul unei discipline înrudite cu biologia celulară, numită „biologie moleculară”. Din această disciplină ne vom opri la două mecanisme:

- replicarea ADN,
- sinteza proteinelor.

4.1. Structura generală a unei celule umane

4.1.1. Enumerarea principalelor componente ale celulei umane

a) membrana,
b) citoplasma,
c) nucleu,
d) organite celulare.

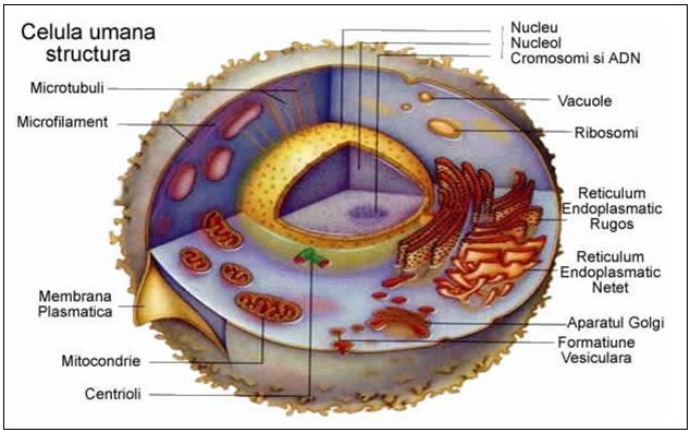


Fig. 4.1.1. Structura celulei umane

4.1.2. Membrana celulară

A. Membrana celulară delimitează celula la exterior, are rol de susținere şi separare, fiind sediul unor procese de schimb de substanțe între celulă şi mediul său extern.

B. Membrana celulară este alcătuită distr-un strat dublu foto-lipidic, conținând în forme şi proporții diferite o serie de structuri proteice (modelul „mozaic lichid" Singer-Nicholson, 1972), (figura 4.1.2.a).

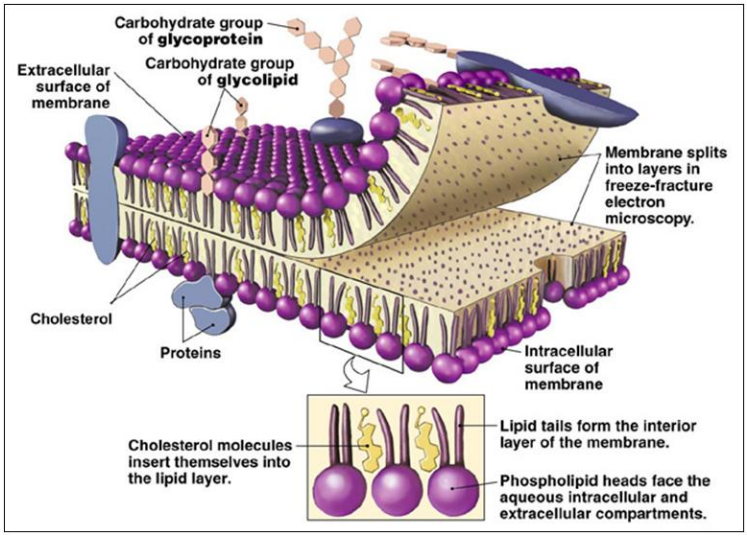


Fig. 4.1.2.a. Membrana celulară-modelul , mozaic lichid"

C. Proprietăți fizice

Membrana celulară are o grosime de 3-10 nm, cu slabă conductibilitate electrică, dar cu mare capacitate electrică, greu permeabilă pentru apă şi ioni.

D. Matricea membranei celulare este alcătuită din fosfolipide ce formează un strat dublu.

Fosfolipidele au caracter amfipatic, având un capăt hidrofil şi un capăt hidrofob (figura 4.1.2.b).

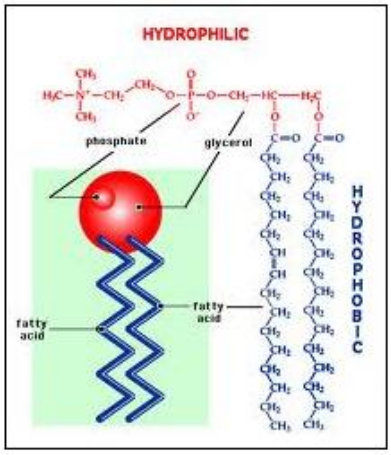


Fig. 4.1.2.b. Structura generală a unui fosfolipid

Acizii graşi au 16-18 atomi de carbon în lanț şi, având caracter hidrofob, se orientează spre interiorul membranei, în timp ce restul fosforic, având grupări OH este hidrofil, fiind orientat spre fețele exterioare ale membranei, fie spre interiorul celulei, fie spre exterior.

E. Proteinele membranare formează „insule" în matricea fosfolipidică. Ele pot fi:

a) proteine intrinseci, care pot să fie la rândul lor:

- proteine ce traversează membrana (canale ionice pentru transport pasiv sau pompe ionice pentru transport activ), figura 4.1.2.c,
- proteine parțial înglobate, ce formează structuri numite „receptori membranari", cu rol foarte important în funcționarea membranei (de receptori se pot lega diferite molecule-hormoni, medicamente etc.).

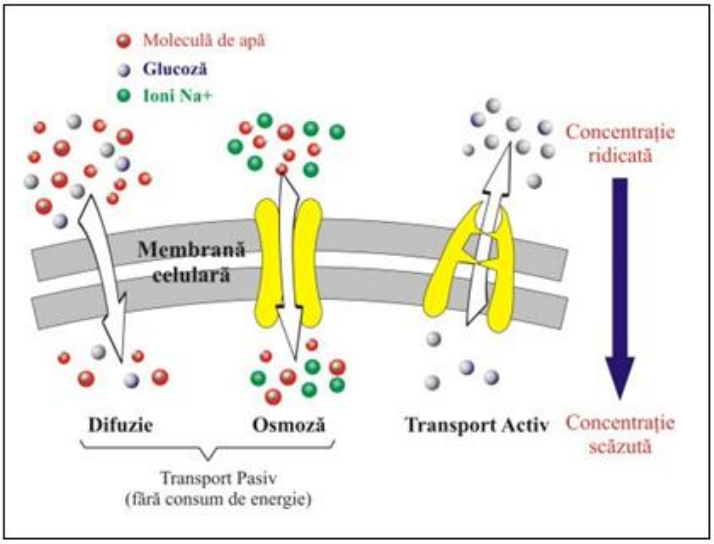


Figura 4.1.2.c. Canale şi pompe ionice

b) Proteine extrinseci, legate slab la suprafața membranei.

F. Membrana celulară este totodată sediul fenomenelor electrice, de producere a potențialului de membrană.

G. Membrana are un rol important în semnalizarea inter şi intracelulară privind transferul de informații în aceste procese de semnalizare sunt implicați şi receptorii membranari.

H. Să încheiem cu rolul membranei în apoptoză - fenomenul de moarte naturală programată.

4.1.3. Citoplasma

A. Citoplasma este o soluție apoasă ce reprezintă mediul de dispersie pentru substanțele dizolvate:

- ioni: [K^+] 150 mM, [Na+] 10 mM, [Cl^-] 100mM,
- molecule mici neutre [glucoză],
- macromolecule (proteine, polizaharide etc).

B. Citoplasma este străbătută de fire ce se sprijină pe membrană, nucleu şi organitele celulare mari, ce formează un citoschelet.

C. Proprietățile citoplasmei sunt sintetizate în tabelul 4.1.3.

Tabel 4.1.3. Proprietățile fizice ale citoplasmei

Ion	Concentration in cytosol (millimolar)	Concentration in blood (millimolar)
Potassium	139	4
Sodium	12	145
Chloride	4	116
Bicarbonate	12	29
Amino acids in proteins	138	9
Magnesium	0.8	1.5
Calcium	<0.0002	1.8

4.1.4. Nucleul celular

A. Nucleul celular este o formațiune sferică sau ovoidală, cu diametrul de 4-6 µm, alcătuit în cea mai mare parte din ADN.

B. Structura nucleului (figura 4.1.4.a)

În interior se găseşte o formațiune mai densă numită „nucleol", iar la exterior este mărginit de o membrană nucleară, cu pori prin care pot trece molecule mici. Transportul prin membrana nucleară (pentru ARN şi proteine) este asigurat de molecule numite „cargo-GTP-aze."

C. Nucleul este sediul de stocare a informației genetice şi joacă un rol important în procesul de diviziune celulară şi în procesul de sinteză a proteinelor.

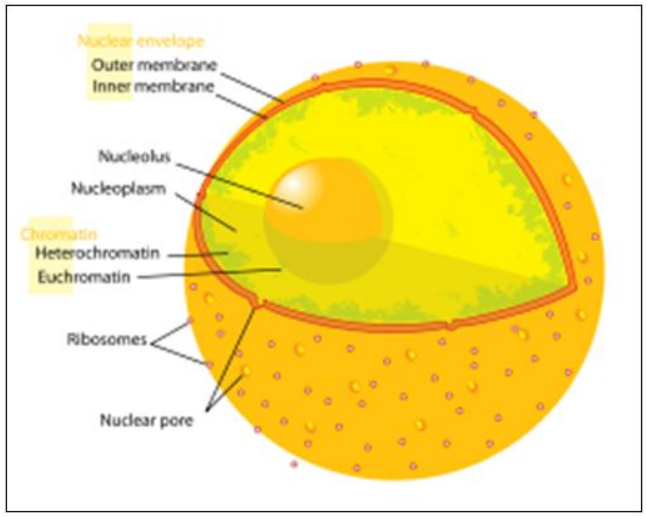


Fig. 4.1.4. Structura nucleului

4.1.5. Organitele celulare

În citoplasmă se găsesc numeroase organite celulare, fiecare cu un rol bine definit în funcționarea celulei:

- reticulul endoplasmic,
- ribozomi,
- mitocondrii,
- aparatul Golgi,

- lizozomi,
- centrioli,
- microtubuli, microfilamente,
- vacuole, vezicule.

Ne vom opri pe scurt doar la ribozomi şi mitocondrii, având în vedere importanţa lor în studiile de bioinformatică.

4.1.6. Mitocondria

A. Mitocondria este un organit celular care este sediul sintezei moleculelor macroergice de ATP.

B. Are formă alungită având în interior „creste" formate prin plierea membranei interne. Membrana internă conține molecule ce asigură un transfer de electroni (NADH, FAD, ubiquinonă şi citocrom c), însoțit de eliminare de protoni. Procesul se numeşte „fosforilare oxidativă", având ca rezultat sinteza unei molecule de ATP. Funcționarea este asigurată de o pompă de protoni descrisă sub numele de" teoria chemiosmotică" de către Mitchell, laureat al premiului Nobel în 1978.

Schematic, transferul electronilor în membrana internă a mitocondriei este prezentată în figura 4.1.6.a, iar principiul pompei de protoni este sintetizat în figura 4.1.6.b.

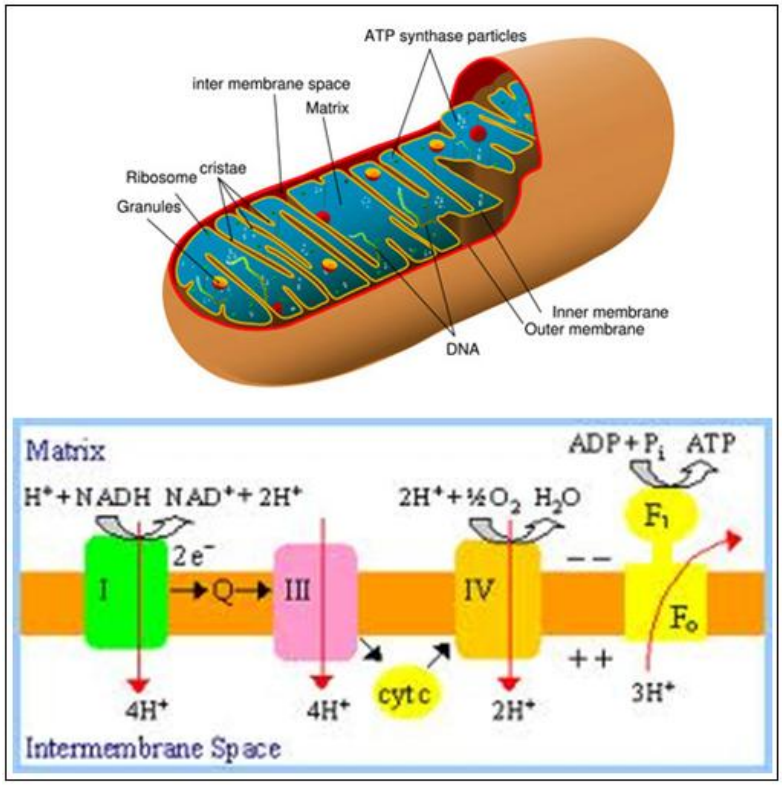


Fig. 4.1.6.a. Mecanismul fosforilării oxidative

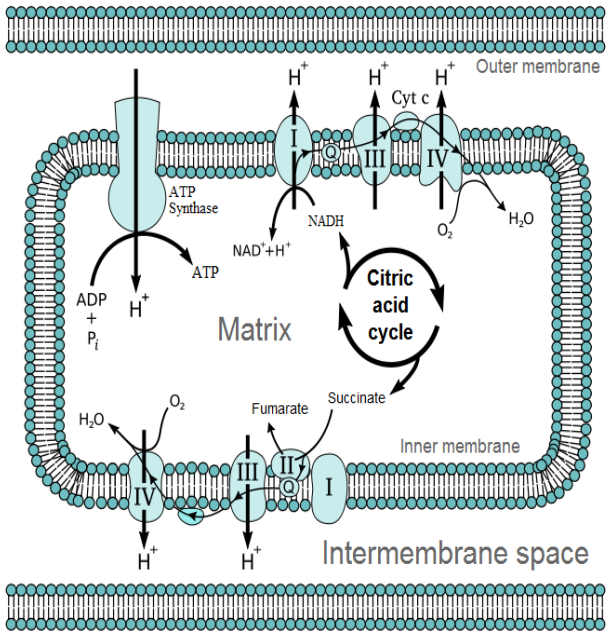


Fig. 4.1.6.b. Teoria chemiosmotică a pompei de protoni

4.1.7. Ribozomii

A. Ribozomii sunt cele mai mici organite celulare, descoperite de către George Palade, de origine română, premiul Nobel în 1974.

B. Sunt alcătuiți din ARN ribozomal şi sunt sediul sintezei proteinelor.

C. Un ribozom are trei situs-uri, notate E, P şi A, în care se poate lega o moleculă de ARNt (ARN "de transport" sau „de transfer", figura 4.1.7.)

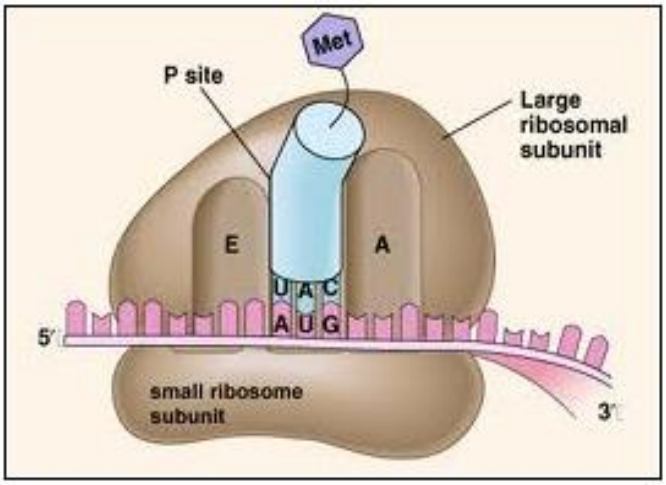


Fig. 4.1.7. Structura ribozomilor

4.2. Diviziunea celulară

Majoritatea celulelor umane (sunt doar unele excepții) se divid în cursul vieții lor. Procesul de diviziune se petrece diferit la celulele somatice, față de cele sexuale şi vom trata separat cele două cazuri.

4.2.1. Mitoza

A. Mitoza este procesul de diviziune al celulelor somatice.

B. Celulele somatice sunt diploide – au fiecare cromozom în două exemplare. În cazul celulelor umane sunt 23 de perechi de cromozomi (unele detalii vor fi prezentate în cursul de genetică).

C. Ciclul de viață al unei celule somatice, adică intervalul de timp dintre două diviziuni, cuprinde mai multe faze, în care se sintetizează la început ARNm şi proteine, apoi se sintetizează şi ADN – practic se dublează cantitatea lui prin procesul de replicare în nucleu.

D. Diviziunea celulară ocupă cca 10% din ciclul celular şi are mai multe faze:

- profaza – cromozomii devin vizibili, fiecare fiind dublat longitudinal, apar 2 centrioli ce migrează spre polii celulei, dispare membrana nucleară şi se formează fusul de diviziune,
- metafaza – cromozomii se aranjează în regiunea ecuatorului şi se leagă de fibrele fusului în zona centromerului,
- anafaza – deplasarea cromatidelor fiecărui cromozom spre polii celulei,
- telofaza – formarea la fiecare pol a câte unui nucleu, cu formarea de membrană nucleară.

În final se divizează şi citoplasma formându-se două celule fiice cu nuclee identice.

Fenomenele descrise mai sus sunt prezentate sintetic în figura 4.2.1

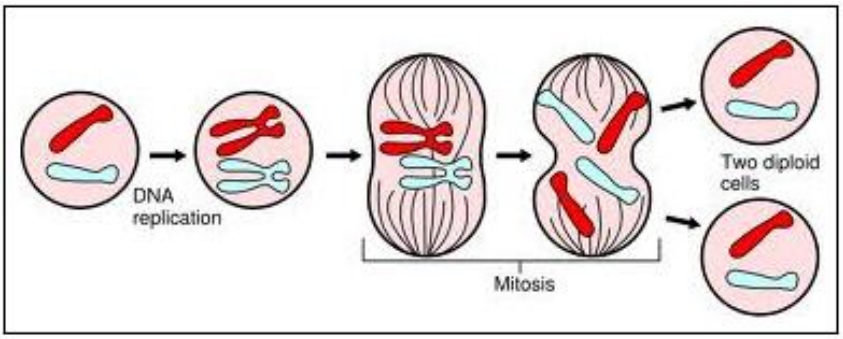


Fig. 4.2.1. Mitoza

4.2.2. Meioza

A. Diviziunea prin meioză este tipică pentru celulele germinale, ovocitul şi spermatocitul, în zona de maturizare ce formează celule sexuale, numite şi „gameți".

B. Meioza cuprinde două diviziuni succesive:

a. diviziunea reducțională – când dintr-o celulă germinală diploidă se formează două celule haploide (cu fiecare cromozom într-un singur exemplar),

b. diviziunea ecuațională – asemănătoare mitozei, cu deosebirea că fiecare celulă care se divide acum este o celulă haploidă.

C. În final rezultă 4 celule haploide. În figura 4.2.2 este reprezentată în partea din dreapta meioza, comparativ cu mitoza reprezentată schematic în partea din stânga, fiind vizibile deosebirile din toate fazele diviziunii.

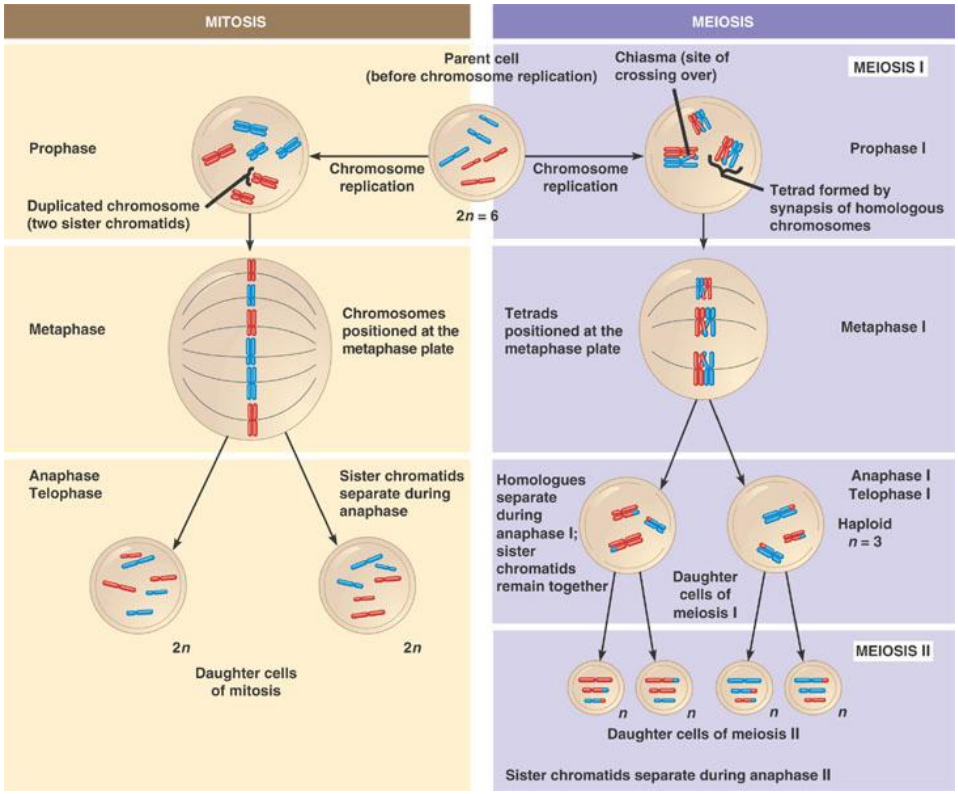


Fig. 4.2.2. Meioza

Să urmărim în continuare două procese esențiale din biologia moleculară: replicarea ADN şi sinteza proteinelor.

4.3. Replicarea ADN

4.3.1. Noțiunea de replicare

A. Descifrarea mecanismului replicării ADN a avut o importanță deosebită prin aportul său la înțelegerea mecanismelor genetice de transmitere ereditară a informației.

B. Replicarea reprezintă procesul de producere a unei „replici", adică o copie identică cu originalul. În cazul ADN este vorba de producere a două molecule de ADN identice pornind de la una singură.

4.3.2. Fazele replicării ADN

A. Replicarea este inițiată într-un punct al dublului helix, numit „origine” - o regiune cu secvență recunoscută de o proteină inițiatoare a replicării aceste regiuni sunt de obicei bogate în perechi AT, mai ușor de desfăcut.

B. Pornind de la origine, dublul helix este despiralat, cu o enzimă – topoizomeraza, apoi este desfăcut asemănător cu un fermoar, fiind rupte toate punțile de hidrogen pe o anumită porțiune din helix de către o enzimă - helicaza. Sunt astfel expuse liber spre exterior bazele azotate, în succesiunea lor, ale ambelor lanțuri ale helixului. Lanțul care este desfăcut de la 3’→5’ încât completarea să apară în succesiunea firească 5’→ 3’ se numeşte „lanț conducător” (”leading strand”), iar celălalt este „lanț întârziat” (”lagging strand”) (figura 4.3.2.)

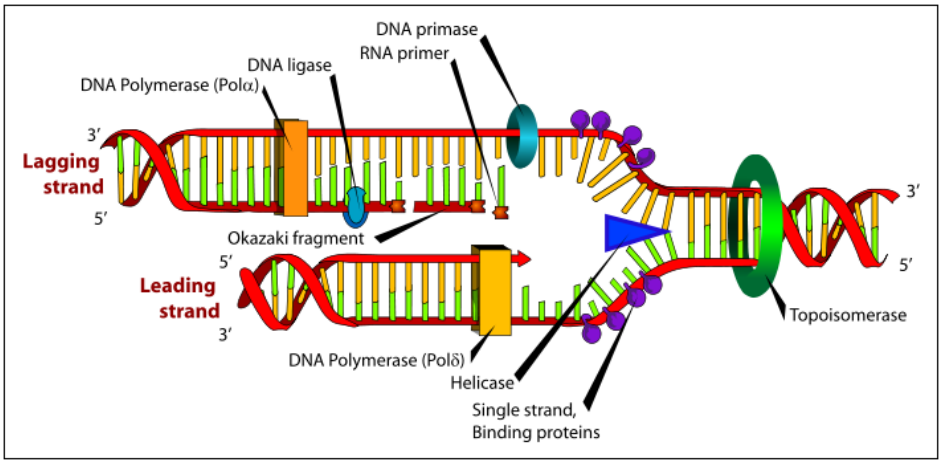


Fig. 4.3.2. Mecanismul replicării ADN

Forma de „furcă” a dublului helix desfăcut de helicază a generat şi numele structurii în această fază: ”furcă de replicare” („replication fork”).

C. Pe lanțul conducător pozițiile expuse sunt ocupate succesiv de către nucleotidele corespunzătoare (conform perechilor posibile A-T şi G-C), proces condus de o enzimă numită ADN-polimerază. Însă sinteza pe celălalt lanț nu poate fi realizată natural, succesiunea de sinteză fiind inversă, de la 3’→5’. De aceea, informația pentru un set de baze azotate libere este întâi transpusă în ordinea posibilă 5’→3’ pe nişte fragmente denumite Okazaki, fragmente ce sunt apoi cuplate pe lanț cu ajutorul enzimei ADN-ligaza.

D. Fiecare din lanțurile inițiale este acum completat cu un lanț pereche, care este spiralat şi devine un dublu helix identic cu originalul.

E. În eucariote procesul de replicare poate începe în mai multe poziții din dublul helixurilor de ADN.

F. Replicarea poate să fie incompletă şi să nu ajungă chiar până la capătul fiecărui cromozom, numit „telomer”, rezultând o scurtare generație după generație, fenomen considerat a avea un rol în procesul de îmbătrânire, în apoptoză sau în apariția unor boli (inclusiv unele forme de cancer).

4.4. Sinteza proteinelor

4.4.1. Paradigma centrală a bioinformaticii

A. Încercările de a explica marea variabilitate biologică fenotipică în contextul unității moleculare a structurilor biologice a condus la plasarea acesteia într-o poziție cheie pentru bioinformatică. Întrebarea la care dorim să răspundem este: Cum este transmisă informația stocată în structura moleculelor de ADN din nucleele celulelor, către celule, pentru a coordona practic toate procesele celulare, dar în special sinteza proteinelor, care sunt atât elemente structurale cât şi elemente de control (enzime) a proceselor celulare. Mecanismul sintezei proteinelor este azi în bună măsură cunoscut, deşi există încă elemente nu pe deplin elucidate. La descifrarea mecanismelor şi-au adus contribuția cercetători din diverse domenii, un rol important avându-l cei din domeniul bioinformaticii.

B. Schema din figura 4.4.1. reprezintă sintetic principalele procese ce au loc pentru sinteza proteinelor.

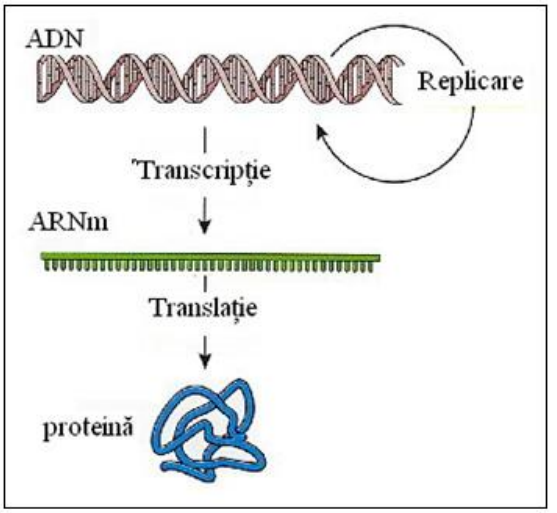


Fig. 4.4.1. Sinteza proteinelor – dogma centrală a bioinformaticii

Vom descrie în continuare aceste procese.

4.4.2. Transcripția

A. Regiunea din ADN care conține informația privind sinteza unei proteine se numeşte genă de sinteză. În amonte față de gena de sinteză se găseşte o genă promotor, cu rol în declanşarea procesului de copiere a informației. Zona promotor mai este numită 5' UTR – 5' untranslated region, iar după gena de sinteză mai este o porțiune ce nu codifică secvență proteică numită 3' UTR.

B. Mecanismul transcripției

La activarea zonei promotor, dublul helix ADN din regiunea genei de sinteză este desfăcut în sensul 3'→5' de către helicază, apoi intervine ARN-polimeraza, care facilitează formarea unui lanț ARNm în sensul normal 5'→3'. Un singur lanț din cele două ale ADN este utilizat pentru citire, numit „lanț matriță" („template strand"), celălalt fiind numit lanț de codificare („coding strand"), deoarece conține secvența exact

în forma în care apare în molecula de ARN sintetizată, cu singura deosebire că, în ARN, în loc de timină apare uracilul. Transcripția este prezentată schematic în figura 4.4.2.

Procesul de transcripție are trei faze: prima este „inițierea", prin activarea promotorului, a doua se numeşte „elongare", reprezentată prin adăugarea, nucleotid cu nucleotid, a componentelor lanțului de ARN, iar ultima fază este „terminarea" (formarea unei bucle bogată în G-C, urmată de o succesiune de U).

Să menționăm că există şi transcripție reversă, adică din ARN în ADN, întâlnită în unele cazuri patologice, cum ar fi infecția cu HIV.

C. Migrarea ARNm

Lanțul ARN format în nucleu se numeşte ARN mesager şi se notează ARNm. În nucleu se mai găsesc nişte organite numite „spliceosomi" care asigură un proces numit „splicing". Prin acest proces sunt eliminate din molecula de ARNm formată, regiunile fără rol în codificarea secvenței proteice, numite „introni", rămânând numai regiunile codificatoare, numite „exoni". Molecula nou formată va traversa membrana nucleară şi va ajunge la ribozomi, unde se va desfăşura procesul de translație.

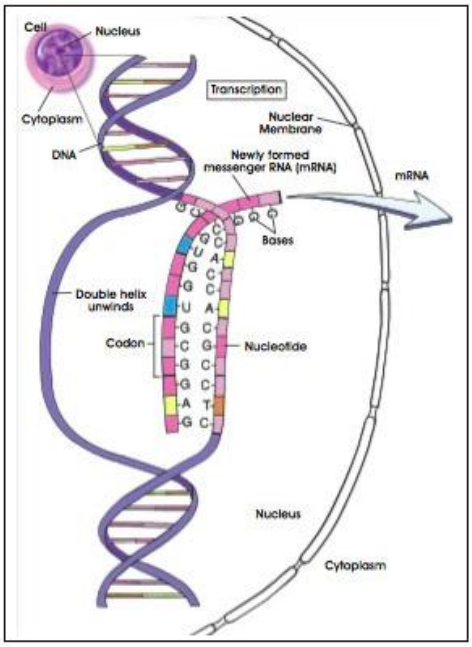


Fig. 4.4.2. Transcripția

4.4.3. Codul genetic

A. Pentru a putea folosi informația purtată într-un alfabet cu 4 litere, cum este cazul acizilor nucleici, la codificarea a 20 aminoacizi, vom avea nevoie de o succesiune de cel puțin 3 litere (George Gamow), care asigură $4^3 = 64$ combinații posibile (cu 2 litere am fi putut obține numai $4^2 = 16$ combinații. Această ipoteză, că o succesiune de 3 baze azotate într-o secvență ADN codifică 1 aminoacid într-o secvență proteică a fost confirmată experimental (Nirenberg, 1961). A fost astfel introdus termenul de „codon", definit ca un triplet de baze azotate într-un lanț de acid nucleic pentru codificarea unui aminoacid într-o secvență polipeptidică.

1st letter	Second Letter: U	Second Letter: C	Second Letter: A	Second Letter: G	3rd letter
U	UUU Phe UUC UUA Leu UUG	UCU UCC Ser UCA UCG	UAU Tyr UAC UAA Stop UAG Stop	UGU Cys UGC UGA Stop UGG Trp	U C A G
C	CUU CUC Leu CUA CUG	CCU CCC Pro CCA CCG	CAU His CAC CAA Gln CAG	CGU CGC Arg CGA CGG	U C A G
A	AUU AUC Ile AUA AUG Met	ACU ACC Thr ACA ACG	AAU Asn AAC AAA Lys AAG	AGU Ser AGC AGA Arg AGG	U C A G
G	GUU GUC Val GUA GUG	GCU GCC Ala GCA GCG	GAU Asp GAC GAA Glu GAG	GGU GGC Gly GGA GGG	U C A G

Fig. 4.4.3. Codul genetic

B. Cercetări ulterioare (Khorana, Holley, premiul Nobel 1968) au permis stabilirea codului genetic prezentat în figura 4.4.3.

C. Observăm că pentru majoritatea aminoacizilor există mai multe codificări posibile. De asemenea există un codon de star, precum şi trei codoni de stop – cărora nu le corespunde nici un aminoacid.

4.4.4. ARN de transport

A. În mecanismul sintezei proteinelor este necesară implicarea unor molecule care să asigure corespondența stabilită în codul genetic. Acestea sunt moleculele numite ARN de transport sau de transfer, notate ARNt (studiate în detaliu de Ochoa - premiul Nobel 1959).

B. Structura ARNt

Molecula de ARNt este destul de mică în comparație cu alte tipuri de ARN. Ea are 4 brațe dintre care două formează legături în timpul translației (figura 4.4.4)

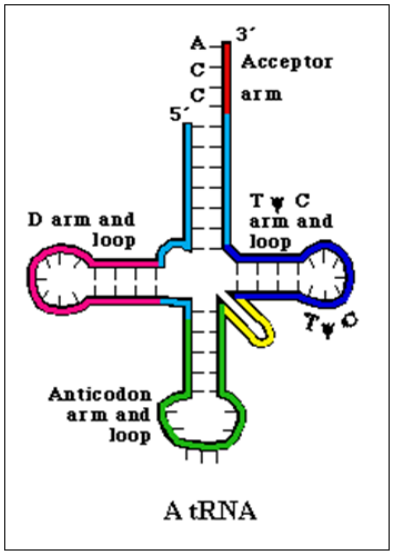


Fig. 4.4.4. Structura moleculei de ARNt

C. Observăm că brațul superior are o porțiune cu capătul 5' liber, care este de obicei fosforilat, iar la capătul 3' se poate ataşa aminoacidul – ataşarea este specifică.

D. Brațul opus are o buclă ce conține o secvență de trei baze azotate complementare codificării aminoacidului, secvență care se numeşte „anticodon". Această porțiune este cea cu care molecula de ARNt se leagă de secvența potrivită din matrița ARNm.

E. Celelalte două ramuri, buclele D şi T facilitează mecanismul de cuplare pe ribozom.

4.4.5. Structura ribozomilor

Revenim la descrierea structurii ribozomilor, începută în 4.1.7.

Ribozomii au două regiuni, o regiune care poate cupla molecule de ARNm venită din nucleu şi o regiune care conține 3 situs-uri, notate E, P şi A, regiune în care se leagă ARNt.

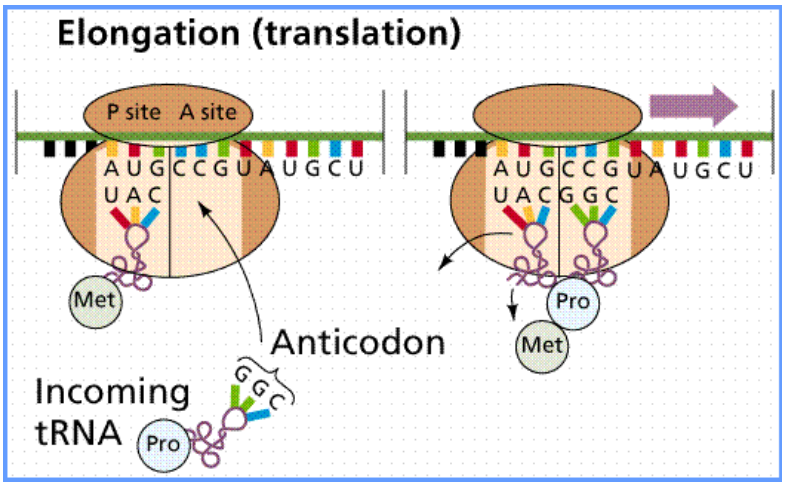


Fig. 4.4.5. Translația

4.4.6. Mecanismul translației

A. Când ARNm expune spre situl P porțiunea cu codonul de start (AUG), situl poate fi ocupat de un ARNt ce codifică metionina „Met" (conform codului genetic).

B. În situl A, încă liber, se poate lega de un ARNt care să aibă anticodonul corespunzător următorului codon din ARNm, codon expus către situl A.

C. Aminoacidul legat de ARNt de pe situl A se cuplează printr-o legătură peptidică de aminoacidul legat de ARNt de pe situl P.

D. La formarea acestei legături peptidice, AA din P se desprinde de ARNt de care era legat şi întreaga structură formată se mută cu un codon spre în față: ARNt din P trece în E, ARNt din A trece în P, cu tot cu aminoacizii legați de el şi în paralel se deplasează şi ARNm în unitatea inferioară. Energia necesară procesului este furnizată de o moleculă de GTP, procesul fiind susținut de o altă proteină numită „factor de elongație"; de fapt, această fază se şi numeşte „elongație".

E. Situl A devenit vacant poate fi ocupat de un alt ARNt cu anticodon corespunzător următorului codon din matrița ARNm. Totodată ARNt de pe situl E se desprinde de lanțul ARNm. Aminoacidul legat de ARNt care a ocupat acum situl A

poate forma o legătură peptidică cu AA din poziția P, urmată de desprinderea acestuia de ARNT care l-a purtat.

Procesul se reia pas cu pas, generându-se un polipeptid, care iese din ribozom.

F. Procesul de translație se încheie când pe matrița ARNM apare un codon de stop, care nu are echivalent în ARNt, deci situl A nu se mai ocupă. În acest moment, lanțul polipeptidic format se desprinde de ribozom.

Fazele sintezei proteinelor sunt prezentate schematic în figura 4.4.6.

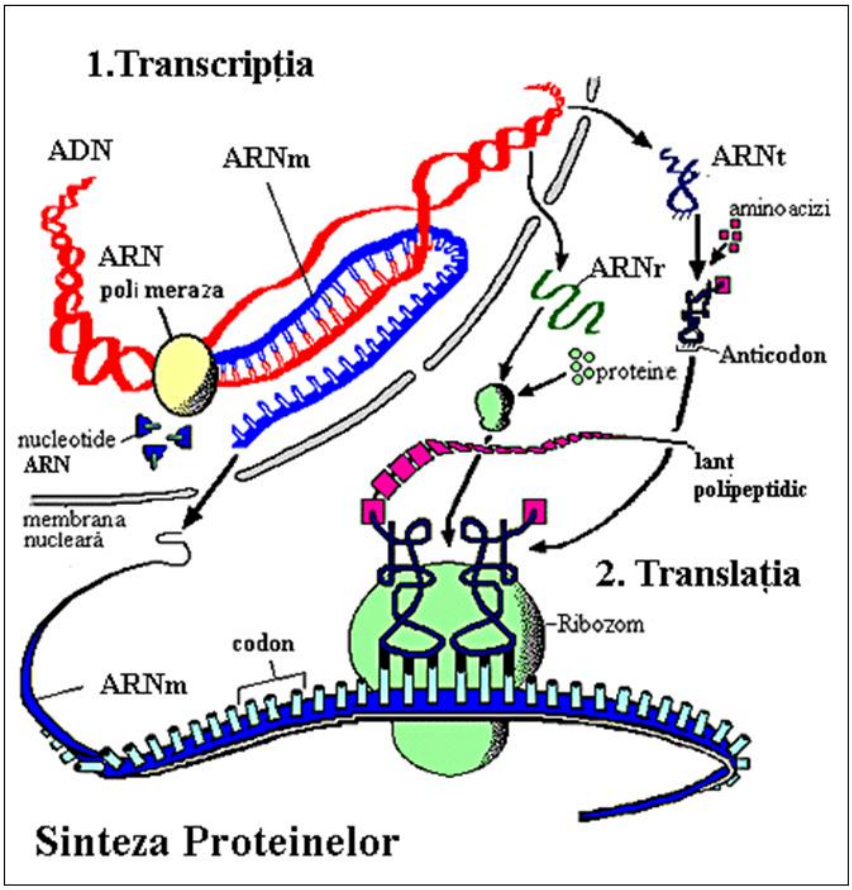


Fig. 4.4.6. Mecanismul translației

4.5. Controlul sintezei proteinelor

Mecanismele moleculare au un sistem de control care dirijează declanşarea, desfăşurarea respectiv terminarea tuturor proceselor.

Controlul sintezei proteinelor la procariote este un mecanism de feedback studiat de Jacob, Monod şi Lwoff (premiul Nobel 1965).

În cazul eucariotelor, mecanismul este mai complicat, descris prin modele cu feedback pozitiv. Nu vom intra aici în detalii privind aceste mecanisme de reglare.

5. Elemente de genetică

A. Genetica este o ramură a biologiei care a cunoscut una dintre cele mai spectaculoase evoluții în ultimele decenii, iar o serie de noțiuni din genetică sunt indispensabile pentru înțelegerea bioinformaticii.

B. Ca obiect de studiu, genetica studiază ereditatea şi variabilitatea organismelor. Diversitatea lumii vii, reprezentată prin diferite caractere ale organismelor, este determinată genetic prin existența unor „factori ereditari" sau „gene" care se modifică sau rearanjează în cursul evoluției.

5.1. Scurt istoric. Capitolele geneticii

5.1.1. Epoca mendeliană

A. Gregor Mendel este recunoscut ca părinte al geneticii; în 1865, în urma studiilor pe diverse soiuri de mazăre, formulează legile eredității şi introduce termenul de „factor ereditar".

B. În 1900 legile lui Mendel sunt redescoperite şi reformulate de Hugo de Vries, Tschermak şi Correus.

C. În 1909, Johanson introduce termenul de „genă" pentru „factorul ereditar".

5.1.2. Teoria cromozomială a eredității

A. La începutul sec. XX, Sutton şi Boveri arată că genele sunt situate în cromozomi.

B. Părintele teoriei cromozomiale a eredității este Morgan, autor a numeroase studii pe Drosophila melanogaster (musculița de oțet), care au condus la localizarea genelor pe cromozomi (devenite ulterior hărți cromozomiale).

C. Sunt identificate tipurile de diviziune, fenomenele de linkage şi cross over şi recombinarea, explicând variabilitatea din natură.

5.1.3. Genetica moleculară

A. Începutul geneticii moleculare a fost reprezentat de descoperirea rolului genetic al ADN (Avery, 1944). În 1953, Watson şi Crick descifrează structura dublu helix a ADN, arătând că genele sunt fragmente de ADN.

B. O suită de descoperiri importante au continuat să se acumuleze, aducând geneticii un palmares important de premii Nobel. Enumerăm aici cele mai semnificative descoperiri: codul genetic, secvențialitatea ADN, reglajul genetic al activității celulare, mecanismul sintezei proteinelor, descifrarea genomului uman.

5.1.4. Bioinformatica

A. De la bun început s-a văzut că funcția principală studiată în genetică este de fapt cea de stocare şi transmitere de informație. S-a conturat astfel o nouă disciplină,

care urmăreşte aceleaşi fenomene ca şi genetica în particular, sau biologia în general, însă din punct de vedere informaţional – bioinformatica. Termenul a fost introdus în 1979 de către Hogeweg.

B. Dezvoltarea bioinformaticii este indisolubil legată de crearea bazelor de date moleculare şi vital asociată de facilităţile oferite de internet.

C. Pentru analiza datelor acumulate, bioinformatica şi-a dezvoltat propriile instrumente de lucru – elaborarea unor sofisticaţi algoritmi de analiză şi comparare a secvenţelor sau a arborilor filogenetici.

D. Biologia sistemelor, ca important capitol al bioinformaticii, tinde să se contureze ca disciplină separată, dedicată proceselor integrative – urmărirea interacţiunilor între componentele sistemelor biologice, pe baza informaţiei purtate de fiecare componentă. În biologia sistemelor se face intensiv uz de metodele de modelare matematică şi simulare pe calculator a proceselor biologice.

5.2. Genetica mendeliană

5.2.1. Terminologie

A. Genetica şi-a dezvoltat o terminologie specifică, ajungându-se cu timpul să se găsească echivalenţii moleculari pentru diverşi termeni. Vom trece aici peste considerentele istorice, adăugând pentru principalii termeni şi explicaţiile moleculare adecvate. În plus, adesea termenii nu vor fi trecuţi ca într-un dicţionar, ci în mod firesc, cum apar în descrierea fenomenelor şi interpretarea rezultatelor. O serie de termeni noi vor fi introduşi ulterior (în secţiunea adecvată).

B. Mendel a lucrat pe plante, în special mazăre (Pisum sativum), urmărind diferite caractere distincte la urmaşi după diferite încrucişări. Câţiva termeni specifici aici:

a) *hibridare* - încrucişarea a doi indivizi ce au una sau mai multe caracteristici diferite. Caracteristicile au adesea manifestări bivalente (în perechi), uneori şi plurivalente. De exemplu: talia plantei: înalte/pitice, culoarea bobului: galben/verde etc. Ulterior s-au găsit explicaţiile moleculare ale multor caracteristici. De exemplu: pentru suprafaţa bobului netedă/zbârcită: prezenţa amidonului generează suprafaţa netedă, conţinut ridicat de dextrină determină suprafaţa zbârcită. În funcţie de numărul caracteristicilor diferite la încrucişare avem:

- *monohibridare* – încrucişare între părinţi cu o singură pereche de caractere diferite,
- *dihibridare* – cazul a două perechi de caractere diferite,
- *polihibridare* – mai multe.

Să mai introducem aici şi termenii:

- *hibrid:* un descendent rezultat în urma unei încrucişări (popular „hibrid”= soi (la plante) sau „rasă” la animale,
- *generaţie:* ansamblul descendenţilor obţinuţi din perechi de părinţi similari.

b) *fenotip* – (termen introdus mai târziu): ansamblul caracteristicilor morfologice, fiziologice, biochimice şi comportamentale ale unui organism.

5.2.2. Legea purităţii gameţilor

A. Observaţiile lui Mendel

a) Prin încrucişarea plantelor (generaţia părinţilor P) cu bob neted, cu plante cu bob zbârcit, în prima generaţie F_1 s-au obţinut numai plante cu bob neted. Acest aspect

(bob neted) al unui caracter (suprafaţa bobului), poartă denumirea de *aspect dominant* (numit şi caracter dominant), notat uzual cu litere mari (A). Aspectul „bob zbârcit", care nu apare în prima generaţie a fost numit „aspect recesiv" (caracter recesiv), notat uzual cu litere mici (a).

b) Prin încrucişarea plantelor din generaţia F_1, în a doua generaţie F_2 apar atât plante cu bob neted cât şi cu bob zbârcit. Raportul între numărul descendenţilor cu aspect dominant (D) şi recesiv (R) a fost [D:R= 3:1].

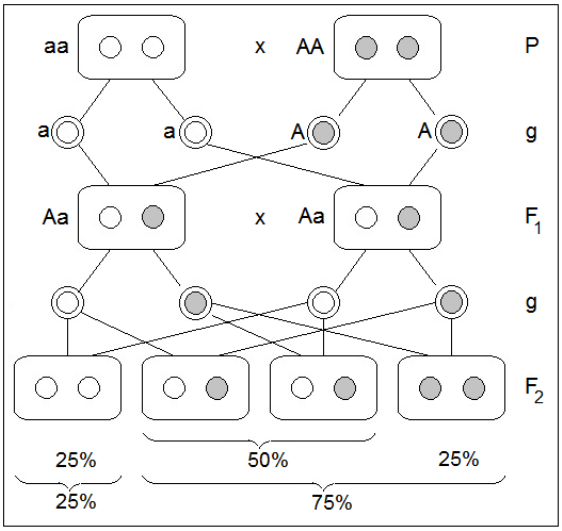


Fig. 5.2.2. Monohibridismul

B. Explicaţia lui Mendel

a) Mendel a intuit că în celulele somatice factorii ereditari se găsesc în pereche, iar în celulele sexuale sub formă simplă, (ulterior dovedit prin numărul de cromozomi: dublu în celulele somatice – celule diploide, simplu în cele sexuale – celule haploide.

b) Celulele sexuale, numite *gameţi*, poartă caracterele părinţilor (în porţiuni numite ulterior „gene" – echivalent cu „factor ereditar" şi aproximativ „1 genă = 1 proteină."

c) Celula-ou, din care se dezvoltă viitorul organism apare prin unirea celor 2 gameţi, unul de la tată şi unul de la mamă.

d) Legea purităţii gameţilor afirmă că: Gameţii sunt puri din punct de vedere genetic, (conţin doar un aspect al unui caracter = factor ereditar). Prin combinarea la întâmplare (probabilistă) a acestor gameţi apare în generaţia a doua fenomenul segregării în proporţia: „3D: 1R".

C. Homozigoţi şi heterozigoţi

a) organismele care au un singur tip de factori ereditari sunt pure din punct de vedere genetic, numite *homozigoţi*. În organismele homozigote, gameţii provenind de la cei doi părinţi poartă acelaşi aspect al unui caracter. Homozigoţii pot fi purtători fie ai aspectului dominant (notaţi AA), fie ai aspectului recesiv (notat aa).

- În observaţiile menţionate anterior s-a presupus că părinţii (P) erau „homozigoţi", aspect justificat prin omogenitatea generaţiilor la încrucişări exclusiv între organisme de acelaşi fel.

b) Organismele impure din punct de vedere genetic, adică cele ce posedau ambele aspecte ale unui caracter – factor ereditar, au primit numele de *heterozigoți*. Însă din punct de vedere al manifestării externe, este vizibil caracterul dominant (figura 5.2.2.a).

c) *Genotip* – totalitatea factorilor ereditari conținuți într-un organism. Termenul devine important în contrast cu „fenotipul", în care surprindem doar aspectul extern, generat de factorii determinanți, deşi acelaşi aspect poate fi generat de genotip-uri diferite.

5.2.3. Dihibridismul şi legea segregării independente a caracterelor

A. Un pas important, efectuat tot de Mendel, a fost analiza aspectelor/caracterelor la descendenți provenind din părinți diferiți prin două perechi de caractere. Pentru abordarea formală să notăm caracterele cu A sau a (dominant = A, recesiv = a, ex.: bob neted A/zbârcit a), respectiv B sau b (bob galben B/bob verde b).

B. În prima generație toate plantele aveau fenotipic boabe netede şi galbene, deşi genotipic o parte erau hibride.

C. În a doua generație au apărut patru feluri de descendenți:
- două caractere dominante – proporție 9/16 (AB)
- un caracter dominant, altul recesiv – proporție 3/ 16 (Ab)
- celălalt caracter dominant, celălalt recesiv – 3/16 (aB)
- ambele recesive – 1/16 (ab).

D. În figura 5.2.3. este prezentată formarea generației F_2 în cazul dihibridismului.

Dihybrid cross

A and *a* represent one trait, and *B* and *b* represent a different trait that is linked to inheritance of *A* or *a*.

	AB	Ab	aB	ab
AB	AABB	AABb	AaBB	AaBb
Ab	AABb	AAbb	AaBb	Aabb
aB	AaBB	AaBb	aaBB	aaBb
ab	AaBb	Aabb	aaBb	aabb

Dominant for *A* and *B* = 9/16
Dominant for *A*, recessive for *b* = 3/16
Recessive for *a*, dominant for *B* = 3/16
Recessive for *a*, recessive for *b* = 1/16

Fig. 5.2.3. Dihibridismul

E. Pe baza constatărilor experimentale, Mendel a enunțat şi legea segregării independente a caracterelor: „Factorii ereditari pereche segregă (se separă) independent de alte perechi de factori ereditari. Raportul de segregare este 3D:1R pentru fiecare pereche, iar în cazul a două perechi este 9:3:3:1".

F. Legea segregării independente a caracterelor a fost susținută de numeroase evidențe experimentale atât în regnul vegetal cât şi animal, venind în sprijinul ipotezei unității lumii vii.

5.2.4. Alte tipuri de segregare

A. Noțiuni de alelă

Conceptului de „factor ereditar", care determină un caracter, i s-a asociat şi termenul de „alelă", căruia i s-a ataşat ulterior şi baza moleculară – gena – reprezentată printr-o porțiune dintr-o moleculă de ADN. Astfel, spunem că aspectul de bob neted sau zbârcit este determinat de o pereche de gene alele, una dominantă, cealaltă recesivă. Organismele homozigote au doar o alelă, sub formă de pereche, fie dominantă (AA), fie recesivă (aa), iar cele heterozigote au ambele gene alele (Aa), fenotipic având aspectul dat de alela dominantă.

B. Relațiile interalelice s-au dovedit mai complexe decât forma simplă dominanță/ recesivitate prezentată mai sus. Amintim aici, fără a intra în detalii, şi alte tipuri de segregare.

a) Dominanța incompletă – destul de frecventă, în care organismele heterozigote nu manifestă fenotipic caracterul dominant ci prezintă o formă intermediară;de exemplu, la planta barba-împăratului (Mirabilis jalapa), la încrucişarea varietății cu flori roşii (AA), cu varietatea cu flori albe (aa), în F_1 apar numai flori roz (Aa), iar în F_2, apar flori roşii/roz/albe în raportul 1:2:1, deci organismele heterozigote prezintă caracterul intermediar.

b) Supradominanța - organismele heterozigote au mai dezvoltate însuşirile biologice (fertilitate, talie, etc), decât părinții homozigoți.

c) Gene letale – gene care în stare homozigotă determină moartea individului înainte de maturitatea sexuală. De exemplu: la încrucişarea a doi şoareci galbeni rezultă şoareci galbeni şi şoareci de altă culoare în raport 2:1, şoarecii galbeni rezultați fiind toți heterozigoți, deci forma homozigotă este letală.

d) Polialelia – acelaşi caracter apare în mai mult de două forme, deci sunt mai mult decât 2 gene alele (A şi a) pentru acelaşi „locus". De ex.: musculițe (Drosophila m) cu ochi albi, roz, purpurii, corai, etc. tipul normal (nemutant) fiind ochi cărămizii.

e) Codominanța - Grupa sanguină la om este determinată de 3 gene polialele: L^A, L^B şi l, unde L^A şi L^B sunt dominante față de l, însă împreună sunt codominante. În tabelul 5.2.4. sunt prezentate schematic genotipurile corespunzătoare.

Tabelul 5.2.4. Fenotipurile şi genotipurile grupelor sanguine la om

Grupa sanguină (fenotip)	**Genotipuri posibile**
AB	$L^A L^B$
A	$L^A L^A$ sau $L^A l$
B	$L^B L^B$ sau $L^B l$
0	ll

Aceste relații sunt utile atât pentru transfuzie cât şi în stabilirea paternității. Cunoscând grupa sanguină a copilului (de ex.: A) şi a mamei (de ex.: 0), se pot stabili grupele sanguine ale tatălui (numai AB sau A).

f) Poligenia – adesea un caracter fenotipic este rezultatul interacțiunii mai multor gene nealele.

5.2.5. Principiul Hardy-Weinberg

A. Genofond

Între indivizii unei populații au loc schimburi de gene. Genotipurile parentale se desfac şi se amestecă la nivelul descendenților, formând un tot unitar la nivelul populației, numit „fond genetic" sau „genofond" - totalitatea genelor unei populații.

B. În 1908 a fost enunțată legea stabilității frecvențelor genice numită şi legea/ principiul Hardy-Weinberg.

a) Enunț: Frecvențele genice rămân constante de la o generație la alta, frecvența genotipurilor fiind şi ea constantă, determinată de frecvența genelor populației parentale.

Legea este valabilă pentru populații panmictice mari (cu încrucişare aleatoare), în care nu acționează forțe modificatoare ale frecvențelor genetice.

b) Din punct de vedere formal, pentru o pereche de gene alele A_1 şi A_2 (nu trebuie precizată dominanța), care au frecvențele p şi q (frecvența ≈ probabilitatea), atunci organismele unei populații pot avea trei genotipuri, A_1A_1, A_1A_2 şi A_2A_2 cu frecvențele p^2, 2pq şi q^2.

Sintetic este prezentat acest lucru în tabelul 5.2.5.

Tabelul 5.2.5. Ilustrarea legii Hardy-Weinberg

♀ \ ♂	A_1 (p)	A_2 (q)
A_1 (p)	A_1A_1 (p^2)	A_1A_2 (pq)
A_2 (q)	A_1A_2 (pq)	A_2A_2 (q^2)

5.3. Teoria cromozomială

5.3.1. Localizarea genelor

Lucrările lui Mendel au fost uitate, principiile sale fiind redescoperite la sfârşitul sec. XIX. De menționat formularea mai clară a lui Hugo de Vries: celulele au câte două copii ale aceluiaşi gene, câte una de la fiecare părinte. La începutul sec. XX s-au acumulat o serie de date privind diviziunea celulară, stabilindu-se legături evidente între datele studiilor de genetică (privind ereditatea) şi localizarea cromozomială a informației genetice.

5.3.2. Morfologia cromozomilor

A. Fiecare specie de viețuitoare are caracteristic numărul şi forma cromozomilor care pot fi vizualizați şi identificați în timpul diviziunii celulare.

B. Un cromozom este alcătuit din două *cromatide*, unite într-un loc numit centromer. Ramurile mai lungi se notează cu „p", iar cele mai scurte cu „q".

C. După poziția centromerului cromozomii pot fi:
- metacentrici – centromerul aproximativ median,
- submetacentrici – centromerul depărtat de mijloc,
- subtelocentrici – destul de aproape de un capăt,
- telocentric – centromerul situat terminal.

D. Cariotip, idiogramă

Ansamblul cromozomilor se numeşte *cariotip* şi este caracteristic fiecărei specii. Dacă se adaugă măsurători pentru fiecare cromozom şi se plasează în ordinea mărimii, se obține „idiograma" speciei respective.

E. Heterozomi, autozomi

Cromozomii somatici poartă denumirea de *autozomi* şi perechile sunt uzual numerotate începând cu cei mai mari.

În afară de cromozomii perechi, celulele conțin şi doi cromozomi ai sexului, denumiți *heterozomi*. Celulele organismelor de sex feminin au o pereche de cromozomi omologi, notați XX, în timp ce cele de sex masculin au un cromozom X şi unul mai mic, notat Z.

F. În timpul meiozei, la formarea gameților, organismele feminine produc doi gameți X, iar cele masculine produc un gamet X şi unul Z.

5.3.3. Transmiterea înlănțuită a genelor (linkage)

A. Drosophila ca model în genetică

Şcoala lui Morgan a utilizat musculița de oțet (Drosophila melanogaster) ca model de studiu, având ciclu de viață potrivit (2 săptămâni/generație), cariotip simplu, (4 perechi de cromozomi), cu cromozomi uriaşi în glandele salivare (uşor de văzut la microscop), fiind cunoscute cca 500 de mutații (modificări ale materialului genetic), însoțite de manifestări fenotipice.

B. Notații

Genele de tip normal (numit şi „sălbatic" se notează cu 1 sau 2 litere urmate de semnul „+", în timp ce alelele corespunzătoare mutante sunt notate fără „+".

Vom folosi în continuare câteva exemple în care am folosit notațiile din tabelul de mai jos 5.3.3.

Tabelul 5.3.3. Exemple de gene alele la Drosophila

Caracterul	**Gene alele** **Forma sălbatică**	**Forma mutantă**
Culoarea ochilor	bw^{+} - ochi roşii [B]	bw ochi maro [b]
Culoarea corpului	b^{+} - gri [C]	b - negru [c]
Forma aripilor	vg^{+} normale [A]	vg – vestigiale (mici) [a]

C. Plasarea liniară a genelor

Este evident că numărul genelor (caracterelor) este mult mai mare decât numărul cromozomilor, deci mai multe gene trebuie să fie plasate pe un cromozom.

O ipoteză centrală în teoria cromozomială a eredității este plasarea lineară a genelor în cromozom.

D. Linkage

Culegând date de la sute de mii de indivizi, de-a lungul mai multor generații, urmărind diferite caractere, s-au constatat numeroase abateri de la legea segregării independente a caracterelor. Au apărut adesea descendenți cu transmiterea unor caractere „în bloc", fenomen numit „linkage". Un exemplu este redat în figura 5.3.3.a, din care reiese că descendenții din generația a 2-a (F_2) sunt doar de două tipuri, în proporție 1:1.

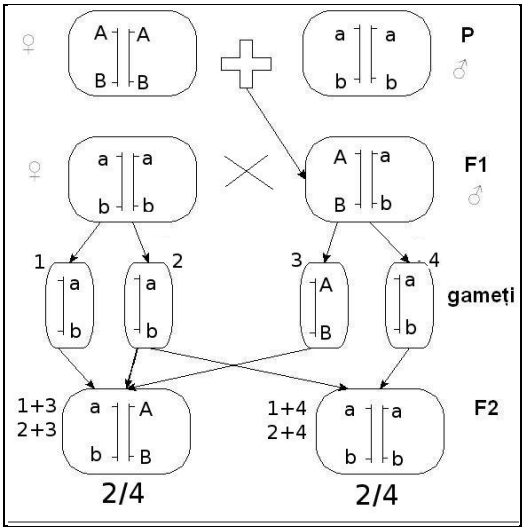


Fig. 5.3.3.a. Evidențierea linkage-ului

Conform legii segregării independente a caracterelor trebuia să avem 4 tipuri de descendenți în proporția 1:3:3:9. (figura 5.3.3.b).

Explicația: genele responsabile de cele două caractere sunt „legate" (se comportă ca una singură), fiind plasate pe acelaşi cromozom (cromozomul 2), apropiate una de alta. În cazul genelor plasate pe cromozomi diferiți legea segregării independente s-a respectat.

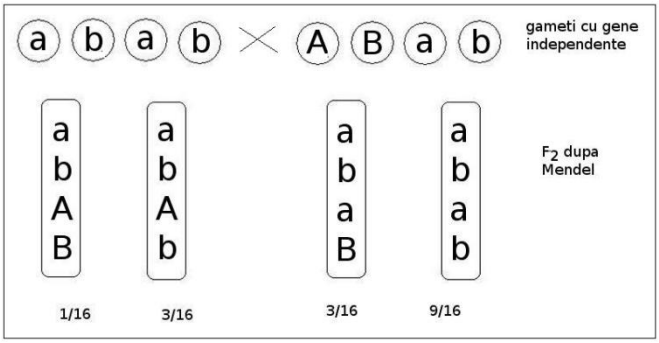


Fig. 5.3.3.b. Varianta mendeliană a exemplului 5.3.3.a

5.3.4. Schimbul reciproc de gene (crossing-over)

A. O altă constatare experimentală importantă a evidențiat că în general, şi fenomenul de linkage are numeroase excepții, deci apar generații cu caractere segregate, dar nu în proporție mendeliană ci mult mai apropiată de cea explicată prin linkage.

B. Un exemplu tipic este redat în figura 5.3.4.a. Din părinți homozigoți mutanți, cu corp mutant negru (cc) şi aripi normale (AA), încrucişați cu musculițe cu corp normal gri (CC) şi aripi vestigiale (aa), apar în prima generație heterozigoți cu fenotip, normal (corp gri, aripi normale). Prin încrucişarea lor cu dubla mutantă homozigotă apar în generația a doua în procent ridicat (41,5%) variantele explicabile prin linkage,

însă apar şi variante în care cele două caractere s-au segregat, în proporție scăzută, trecute în figură în dreptunghiuri cu linie punctată.

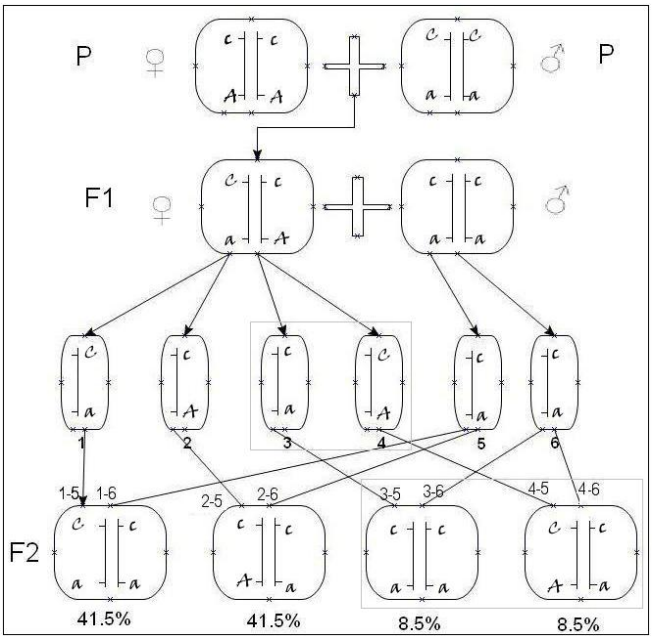


Fig. 5.3.4.a. Exemplu de crossingover

C. Crossing-over

Explicația lui Morgan, dovedită ulterior experimental este următoarea: în timpul meiozei cromozomii omologi se apropie mult, se ating, în unul sau mai multe puncte, iar la separare îşi pot schimba între ei anumite segmente cromatidice, ce duce la schimb de gene.

Ilustrarea acestui fenomen este dată în figura 5.3.4.b.

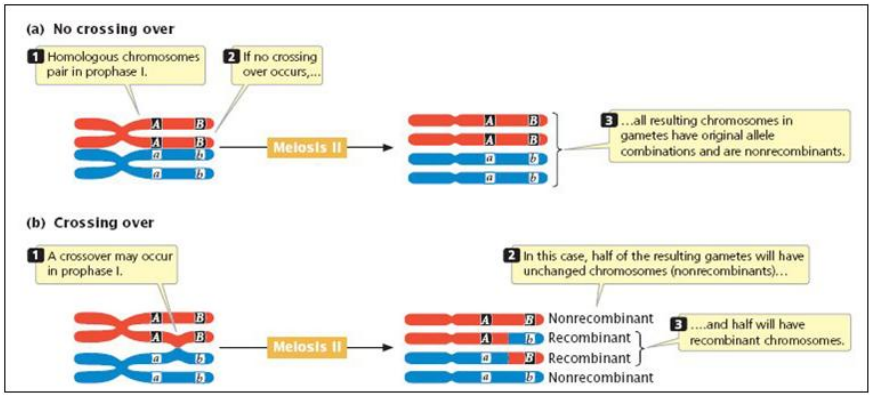


Fig. 5.3.4.b. Schema procesului de crossing-over

Fenomenul de crossing-over poate avea loc şi prin mai multe puncte de contact: crossing-over dublu, triplu sau multiplu, rezultând prin recombinare numeroase variante de rearanjare a genelor.

5.3.5. *Hărți cromozomiale*

A. Procent de recombinare

Un pas important a fost făcut când s-au analizat valorile numerice ale abaterilor de la linkage-ul total. Aceste abateri, explicate prin fenomenul de crossing-over, reprezintă procentul în care poate avea loc procesul de recombinare, notat uzual cu RF („recombination frequency").

B. Distanța între gene

Dacă se face ipoteza că probabilitatea crossing-over-ului este uniformă de-a lungul unui cromozom, atunci putem considera că în cazul a două gene mai depărtate între ele pe cromozom, probabilitatea de recombinare este mai mare. A apărut astfel ideea de a aprecia distanța dintre gene în funcție de frecvența recombinărilor RF.

C. Unitatea pentru distanță genetică

S-a propus numele de „morgan" pentru unitatea fundamentală de distanță genetică, cu submultiplul practic centimorgan (cM) sau m. u. (genetic map unit).

Definiție: O unitate de hartă genetică (1 m. u. = 1cM este distanța dintre două gene cu frecvența de recombinare 1/100 (1 produs din 100 meioze este recombinat).

D. Pe baza datelor experimentale culese s-au putut realiza scheme cu poziția relativă a genelor pe cromozomi. Prima hartă cromozomială a fost realizată de Sturdevant.

În figura 5.3.5 este prezentată o hartă cromozomială simplă rezultată din studii de recombinare.

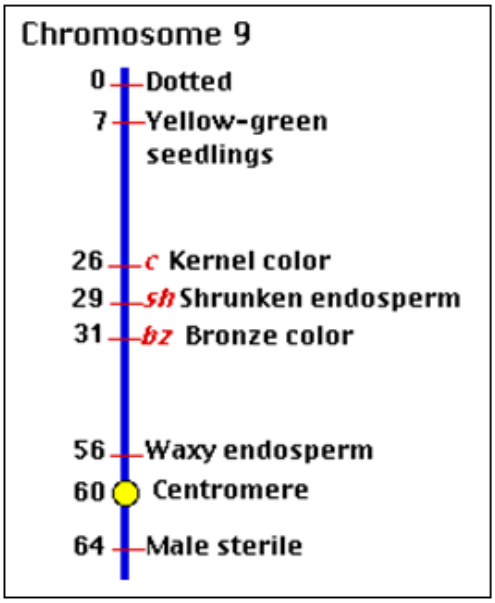


Fig. 5.3.5. Harta cromozomială a cromozomului 9 de șoarece

E. Scoruri pentru RF

Din punct de vedere formal, putem estima frecvența de recombinare printr-un scor numit LOD (Logarithm of ODDS), dat de relația:

$$LOD = Z = log_{10} \frac{probabilitatea\ unei\ secvente\ pentru\ o\ valoare\ data\ a\ linkage-ului}{Probabilitatea\ secventei\ fara\ linkage}$$

$$LOD = log_{10} \frac{(1-\vartheta)^{NR} \times \vartheta^{R}}{0.5^{(NR+R)}} \qquad (5.3.5.a)$$

unde am folosit notațiile:
NR – numărul de urmaşi ne-recombinanți,
R – numărul de urmaşi recombinanți
ϑ - fracția de recombinare [= R / (NR + R)]

5.3.6. Markeri pentru hărți genetice

A. Metodele folosite pentru primele hărți au aplicabilitate limitată, bazându-se pe identificarea vizuală (eventual cu microscopul) a variațiilor fenotipice. În plus, s-a găsit că un caracter poate fi afectat de mai multe gene.

B. De aceea s-a recurs la metode biochimice pentru identificarea diferitelor fenotipuri. Moleculele utilizate pentru astfel de identificări se numesc *markeri.* Multe din rezultatele actuale au fost obținute folosind markeri ADN (aspecte mapate care nu sunt gene).

5.4. Genetica moleculară

5.4.1. Corespondențe moleculare

A. După descoperirea ADN şi a structurii sale, interesul s-a mutat spre determinarea corespondențelor între genele determinate prin metode genetice şi secvențele corespunzătoare din moleculele de ADN.

B. De importanţă covârşitoare a fost dezvoltarea metodelor de determinare a secvențelor nucleotidelor în acizii nucleici, respectiv a secvențelor aminoacizilor în proteine. Acestea au condus în final la stabilirea unor hărți genetice detaliate, cu localizarea genelor responsabile pentru anumite caractere, inclusiv în cazul unor boli genetice.

5.4.2. Structura unui cromozom

A. Reunind datele din biologia celulară şi moleculară cu cele din genetică s-a putut reprezenta schematic structura generală a unui cromozom (figura 5.4.2.).

5.4.3. Hărți fizice

A. Hărțile construite prin metode genetice arată localizările pe cromozomi a genelor pe baza evidențelor fenotipice sau utilizând markeri ADN. Ele au însă un grad limitat de rezoluție şi o acuratețe de asemenea limitată.

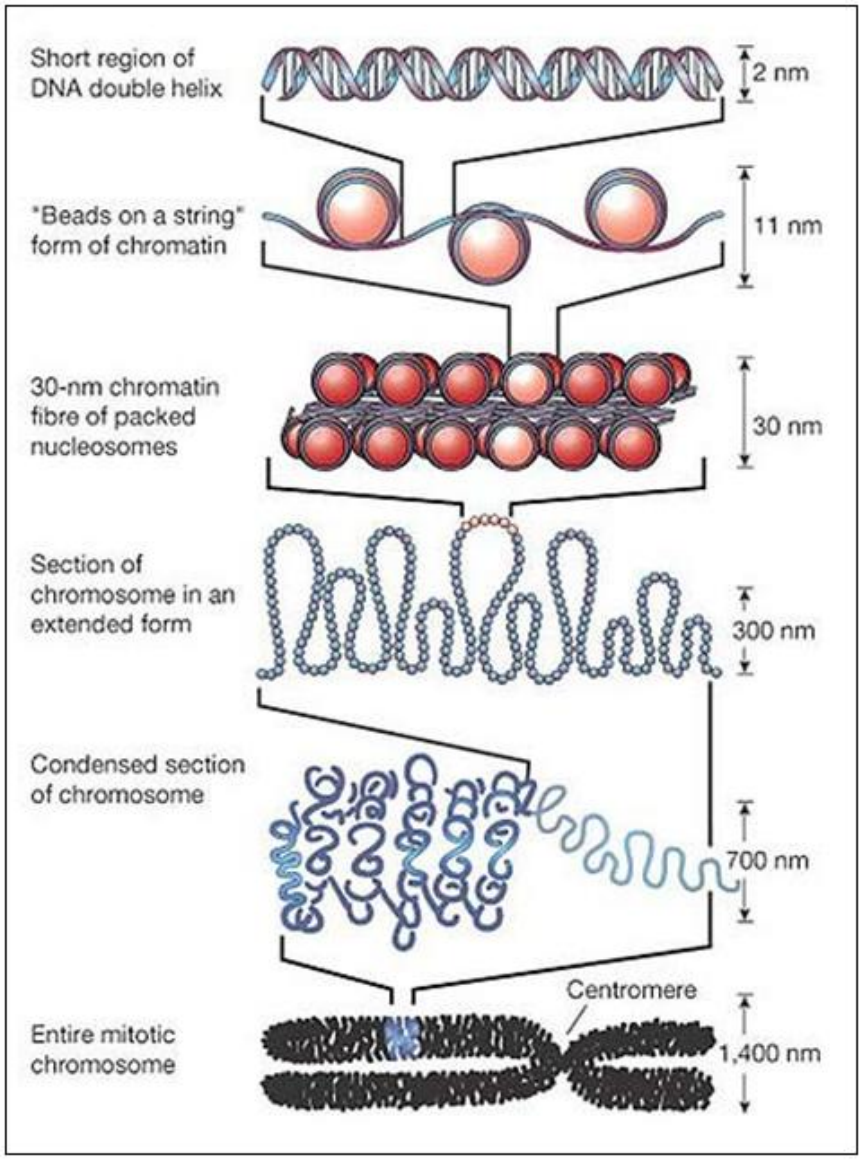


Fig. 5.4.2. Structura unui cromozom

B. O metodă alternativă este construcția unor hărți „fizice" bazate pe date de biologie moleculară, prin analiza directă a moleculelor de ADN. Menționăm existența a trei categorii de metode:

- „restriction mapping", prin localizarea punctelor de acțiune al enzimelor de restricție (endonucleaze de restricție),
- hibridizarea fluorescentă *in situ* (FISH fluorescent în situ hybridization),
- „sequence tagged site mapping" (STS) - se mapează pozițiile unor secvențe scurte prin analiza fragmentelor de genom.

5.4.4. Exemplu de hartă cromozomială

A. Prezentăm ilustrativ în figura 5.4.4. un exemplu de hartă cromozomială.

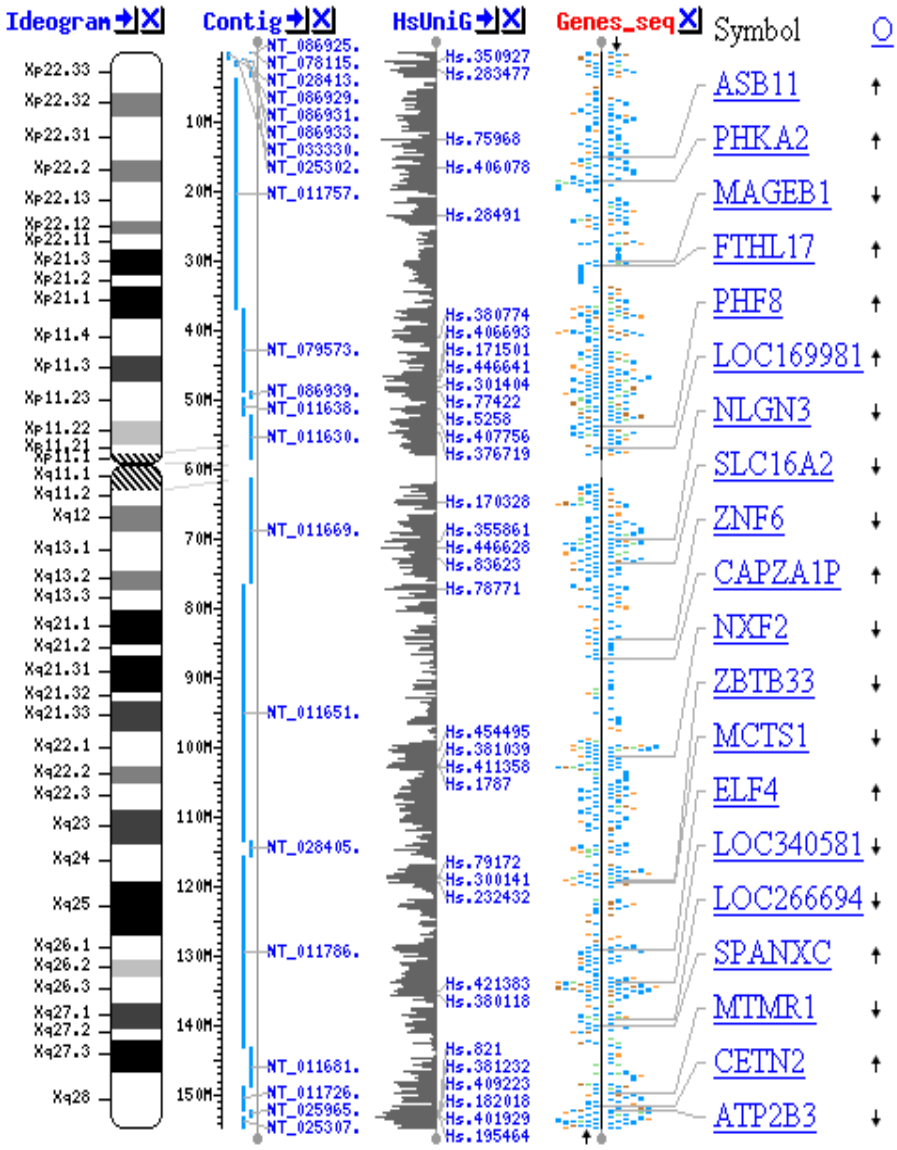


Fig. 5.4.4. Hartă cromozomială

B. În următoarele capitole vor fi prezentate metodele specifice de bioinformatică pentru analiza secvenţelor şi interpretarea structurilor din punct de vedere informațional.

6. Analiza secvențială individuală

6.1. Introducere

6.1.1. Obiectul bioinformaticii

A. Cercetările biologice, care au cunoscut o puternică dezvoltare în ultimele decenii, au condus la acumularea unei cantități imense de informație (sub formă de date), mai ales în domeniul biologiei moleculare. Acum marea provocare este de da sens acestor informații, adică transformarea „datelor" în „cunoştințe", care să ne facă capabili să interpretăm datele. Acesta este chiar obiectul bioinformaticii. Din multitudinea de definiții considerăm cea mai potrivită cea din glosarul NIH.

B. *Definiție:* „Bioinformatica este studiul structurii informației biologice şi a sistemelor biologice. Ea aduce împreună datele de cercetare gnomică cu teoria şi instrumentele matematicii şi a ştiinței calculatoarelor".

6.1.2. Capitolele bioinformaticii

A. Există mai multe posibilități de a privi structural bioinformatica şi a o diviza în capitole – fie în funcție de moleculele studiate, în funcție de aspectele analizate sau în funcție de metodele folosite.

B. Vom aborda aici o clasificare hibridă combinând „aspecte/metode", conform uzanțelor:

a) Baze de date moleculare (se vor trata la lucrări practice)
b) Analiza secvențială (cap. 6, 7, 8 şi 9 precum şi la lucrări practice)
c) Lanțuri Markov (cap. 10 şi 11 precum şi la lucrări practice)
d) Analiza filogenetică (cap. 12 precum şi la lucrări practice)
e) Structuri tridimensionale (se vor trata la lucrări practice)
f) Biologia sistemelor (nu a fost inclusă în acest manual).

6.1.3. Obiectul analizei secvențiale

A. poziție centrală în bioinformatică este ocupată de analiza secvențială. Într-adevăr, confirmarea ipotezei că informația biologică este înregistrată sub forma unor secvențe în structura acizilor nucleici şi a proteinelor, a adus în prim plan metodele de analiză a acestor secvențe.

B. Prin analiză secvențială putem aborda:
- analiza secvențială individuală,
- compararea a două secvențe,
- alinierea multiplă.

6.2. Analiza secvențială grafică

6.2.1. Baze teoretice

A. Proprietățile AA sunt diferite.

Evident, putem extrage o serie de informații privind proprietățile unei macromolecule – aici ne gândim în primul rând la proteine – pornind de la compoziția şi structura sa, având în vedere faptul că elementele structurale – aminoacizii (AA) – au proprietăți diferite:

- reziduurile (catenele laterale) pot fi hidrofile sau hidrofobe,
- dimensiunea,
- sarcina electrică (aceasta depinde şi de pH-ul mediului),
- flexibilitatea legăturilor.

Pentru reprezentări grafice se utilizează în practică reprezentarea unei proprietăți de-a lungul unei secvențe. Prezența au absența unei periodicități este vizibilă în astfel de reprezentări.

Pentru o mai uşoară evidențiere a unor posibile periodicități se folosesc şi ferestre de netezire (în care un punct de grafic reprezintă media unei regiuni în jurul său). Practic s-au dovedit utile ferestre de 9 – 15 elemente. Un astfel de grafic este prezentată în figura 6.2.1.

Nishikawa face o comparație între evoluția sistemelor nevii, dictată de condițiile externe şi a celor vii, dictată de informație.

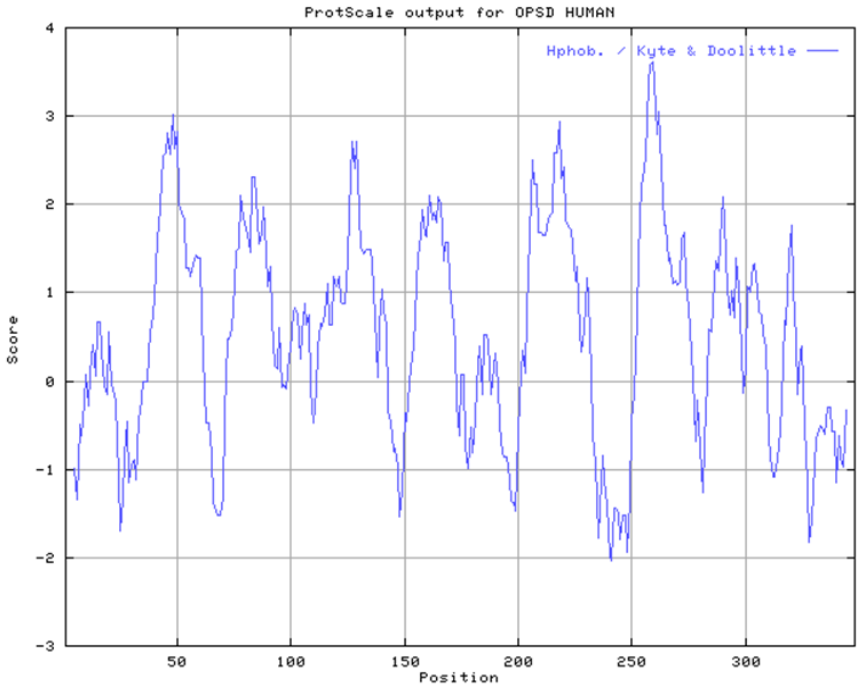


Fig. 6.2.1. Hidrofobicitatea rodopsinei umane (AC P08100) creat cu ExPASy Service Protscale. S-a folosit o secvență de 350 AA şi o fereastră de 9 unități, pe scala de hidrofobicitate Kyte-Doolittle.

B. Analiza proporţiilor AA – există o apariţie preferenţială a AA în anumite structuri secundare.

Aşa cum am văzut în capitolul privind „Structura proteinelor", structurile tridimensionale, începând cu structura secundară depind de structura primară, adică de *secvenţa* aminoacizilor. Pornind de la distribuţia statistică a AA în diferite structuri secundare, s-a constatat o distribuţie preferenţială a AA, dictată de fapt de configuraţia spaţială a atomilor fiecărui AA. Iată câteva exemple:

- ***Glu*** se găseşte aproape exclusiv în *α – helix*
- ***Val (Pro, Tyr***) se găseşte în principal în fâşii *β*
- ***Gly*** este numită şi „spărgător de elice", iar ***Pro*** este de asemenea rar întâlnită în *α – helix* (***Ala, Cys, Leu*** = promotori de helix).

6.2.2. Aspecte structurale

A. AA hidrofobi tind să apară în interiorul proteinelor globulare, iar cei hidrofili spre exterior

- în proteinele membranare regiunile de înserare în membrane, regiunile de înserare în membrană sunt de asemenea bogate AA hidrofobi (având în vedere natura lipidică a moleculelor membranare).

B. Detectarea anumitor periodicităţi ale hidrofobicităţii poate de asemenea sugera apartenenţa la o structură elice sau fâşie.

O aplicaţie pornind de aici ar fi predicţia epitopilor antigeni, prin relaţia lor cu suprafaţa expusă.

6.3. Structura tridimensională

6.3.1. Roţi elicoidale şi elice amfipatice

A. Aranjarea aminoacizilor cu anumite proprietăţi (de exemplu hidrofobicitatea) în structura secundară ne poate sugera proprietăţi şi chiar funcţii ale proteinei respective. S-au găsit astfel de aranjamente în cazul proteinelor structurale, în special pori sau canale transmembranare.

B. bună metodă de vizualizare a regiunilor elicoidale o reprezintă imaginea structurală a unei secvenţe proiectată de-a lungul axei (vedere „din avion"), numită „roata elicoidală". Un astfel de exemplu este redat în figura 6.3.1.

C. În partea de sus, (A), este reprezentată secvenţa de 18 AA care poate forma un *α – helix*. Luând ca punct de plecare pentru reprezentare „poziţia orei 12", plecând în sens orar, sunt trecuţi toţi aminoacizii din secvenţă. Cum o spiră *α – helix* cuprinde cca 3,6 aminoacizi, unghiul între doi aminoacizi succesivi va fi cca 100^0, deci vom avea un aminoacid pe aceeaşi generatoare după o secvenţă de 18 reziduuri (adică 5 spire). Observăm că jumătate din dreapta elicei cuprinde doar aminoacizi nepolari, în timp ce cea din stânga are doar aminoacizi polari. Această aranjare permite agregarea moleculei noastre cu alte suprafeţe hidrofobe şi să poată funcţiona ca por sau canal în membrană.

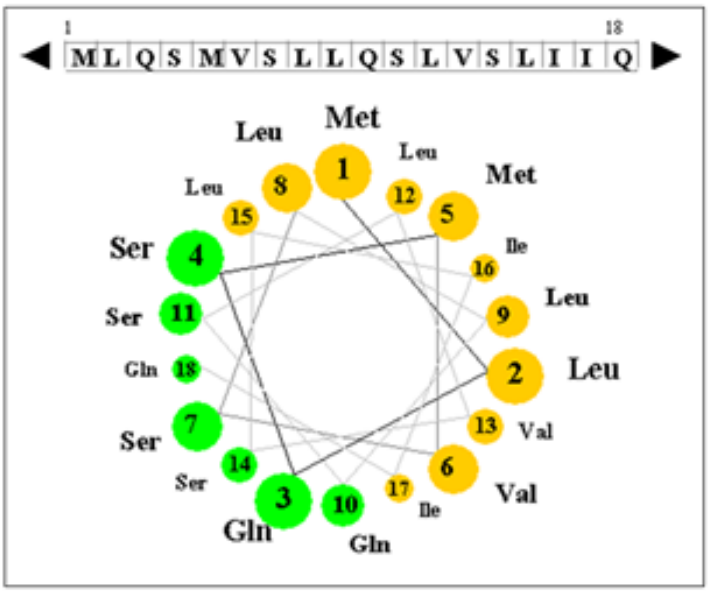


Fig. 6.3.1. O porțiune elicoidală dintr-un lanț polipeptidic. Reziduurile nepolare sunt mai deschise la culoare (galben), cele polare sunt închise la culoare (verzi).
[După O'Neil & Grisham]

6.3.2. Alte structuri

A. În acest mod pot fi depistate unele porțiuni specifice – elice amfipatice (numite şi "coiled-coil"), cu o porțiune hidrofilă şi una hidrofobă, cu succesiuni (3-6 grupe) de câte 7 AA (heptade). Tot aşa sunt detectate şi "fermoarele de leucină" (leucine zipper), domenii ce se leagă de ADN în proteinele reglatoare (factori de transcriere).

B. Trebuie totuşi menționat că există numeroase interacțiuni în cadrul structurii secundare, există o serie de corelații între sub grupuri, astfel încât informația reală extrasă din analiza secvențelor individuale (ca estimare – proprietăților şi rolului proteinelor) nu este pe măsura numărului mare de parametrii analizați.

7. Compararea a două secvențe

7.1. Introducere

7.1.1. Fundamentele analizei secvențiale

A. Observațiile noastre privind lumea vie în mod frecvent bazate pe comparații, găsind adesea asemănări sau deosebiri între diferite structuri sau între diferite funcții. Pentru o abordare științifică, s-a simțit, desigur, nevoia de a introduce metode care să permită o exprimare cantitativă a concluziilor comparației, să putem să evaluăm „cât de mult" seamănă (sau se deosebesc) două (sau mai multe) structuri.

B. Încă de la început ne dăm seama că o comparație reală este multiaxială – putem compara diferite aspecte: forme, dimensiuni, proprietăți de tot felul etc. În plus, un anumit grad de arbitrar (sau convențional) va fi mereu prezent, pentru a grada diverse nivele de deosebiri. Însă, chiar dacă instrumentele pe care le construim sunt imperfecte, ele constituie un suport puternic în interpretarea cantităților imense de informații care însoțesc studiile materiei vii.

C. Metodele elaborate pentru analiza comparativă a două secvențe s-au dovedit a ocupa o poziție centrală în dezvoltarea bioinformaticii, motiv pentru care acestui capitol i se acordă o deosebită importanță în toate manualele și tratatele de bioinformatică.

7.1.2. Evenimente

A. Când comparăm două secvențe urmărim adesea evidențe că ar fi putut proveni dintr-un ancestor comun. Din acesta structurile diverg prin procese de mutație și selecție. „Evenimentele" sau procesele mutaționale de bază sunt:

- substituții: un element (nucleotid în cazul acizilor nucleici, respectiv aminoacid în cazul proteinelor) este înlocuit cu altul,
- inserții: unul sau mai multe elemente sunt introduse într-o secvență,
- deleții: unul sau mai multe elemente sunt eliminate dintr-o secvență.

B. În cazul comparației a două secvențe, unei inserții în secvența X îi corespunde în secvența Y un loc gol, notat „ – „ și numit „gap". În cazul unei deleții din secvența X, vom plasa „gap"-urile în secvența X, ele având corespondent real Y. Vedem deci că inserțiile și delețiile sunt tratate similar, motiv pentru care s-a introdus și termenul „indel".

7.1.3. Termeni

A. În analiza secvențială putem întâlni diferite variante de "asemănări" (similaritate), pentru care se folosesc termini specifici.

Termenul ce desemnează similaritatea se folosește ca adjectiv și poate fi:

a) *homolog* (omolog):
- când secvențele provin din 2 organisme diferite, cu ancestor comun

b) *ortolog*:
- sunt diferențe datorită speciației

- se reține funcționalitatea în evoluție

c) *paralog*:

- din 1 organism, evenimente apărute la duplicare

d) *xenolog*:

- nu au aceeaşi origine prin evoluție
- apar prin evenimente „orizontale” (simbioză, viruşi etc.)

B. Să remarcăm că „similaritatea” ≠ „homologie”.

7.2. Graficele de puncte „dot plots”

7.2.1. Principiul metodei

Este o metodă simplă şi poate cea mai veche [Maizel şi Lenk].

A. *Definiție*: un grafic de puncte (dot plot) este o reprezentare vizuală a similarității între două secvențe, în care cele două secvențe se plasează pe cele două axe ale unei matrice sau ale unui grafic şi se marchează prin puncte toate elementele comune.

B. În forma primară („fără filtre”), un grafic de puncte pentru compararea a două secvențe de acizi nucleici (aceste secvențe conțin doar 4 simboluri), vor apărea multe „zgomote” – coincidențe întâmplătoare, datorită repetitivității simbolurilor.

7.2.2. Filtre

A. De aceea, pentru eliminarea sau reducerea acestor zgomote, se pot aplica „filtre” de diferite lungimi. Astfel, un „filtru de 2 elemente” va permite marcarea unei coincidențe numai dacă regăsim în ambele secvențe un element împreună cu succesorul său (adică se identifică şi se marchează toate grupele de câte 2 simboluri). Putem extinde fereastra la 3 sau mai multe elemente, ajungând să evidențiem într-adevăr anumite secvențe care apar în ambele structuri.

B. În cazul graficelor aplicate pentru proteine, zgomotele sunt mai rare (aceste secvențe conțin 20 de simboluri), în plus, putem introduce chiar scări cu diferențe nuanțe de gri în funcție de rezultatul comparării regiunilor din ferestrele preluate din cele două secvențe. Vom reveni la aceste reprezentări după ce vom prezenta primele „modele de scor”.

C. În figura 7.2.2.a se prezintă graficul de puncte între secvențele X şi Y, fără filtru iar în 7.2.2.b este aplicat un filtru de două elemente pentru aceleaşi secvențe.

D. Graficele de puncte sunt foarte sugestive, fiind recomandate ca un pas inițial de analiză. Din grafic se observă dacă va fi vorba de o similaritate globală sau locală, dacă sunt repetiții sau inversiuni etc. Se folosesc în acest sens şi grafice ale unei secvențe comparate cu ea însăşi, în care, pe lângă diagonala care apare evident, se pot evidenția unele repetiții.

E. Există software dedicat pentru dot-plots: programul „dotter” ce poate fi descărcat de pe serverul ftp EBI.

a) b)

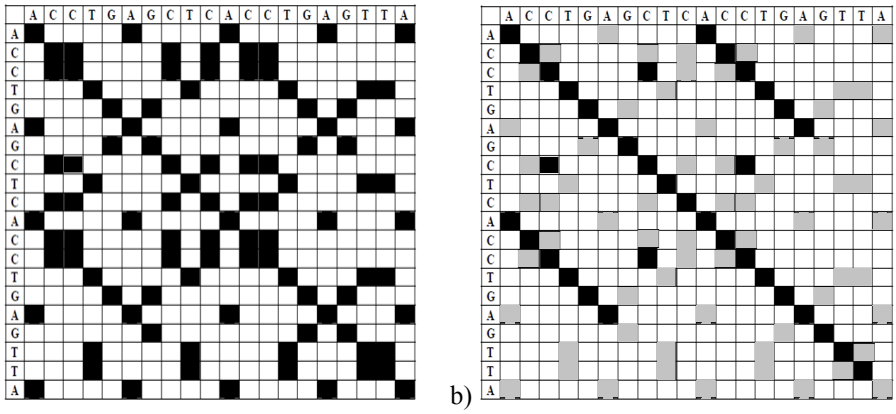


Fig. 7.2.2.: a – dot plot fără filtru, b – dot plot cu filtru de 2

7.3. „Distanțe" între secvențe

7.3.1. Noțiunea de aliniere

Vizualizarea prin „dot-plots", deşi este foarte sugestivă, nu caracterizează cantitativ gradul de similaritate sau deosebire între secvențe. În acest sens s-au propus varii metode de estimare cantitativă a asemănării între secvențe, pe care le vom studia în continuare.

A. Toate pornesc de la **alinierea secvențelor**, pe care o vom defini aici.

Definiție: Alinierea secvențelor (sequence alignment) este o metodă de comparare a două (sau mai multe secvențe) prin scrierea lor una sub alta (element) cu scopul de a evalua similaritatea între aceste secvențe.

În acest capitol ne vom limita la alinierea a două secvențe, urmând apoi a generaliza metoda pentru aliniere multiplă.

Situațiile cu care ne putem întâlni sunt următoarele:
- dacă acelaşi simbol apare în ambele secvențe vom considera că poziția s-a conservat în evoluție,
- dacă simbolurile sunt diferite, se presupune că cele două derivă dintr-un ancestor comun, care poate fi unul din cele două sau chiar altul decât cele două,
- secvențele comparate pot avea lungimi diferite datorită posibilelor inserții sau deleții.

B. Alinierea globală şi locală

În cazul în care comparăm două secvențe în totalitatea lor, vorbim de **aliniere globală**.

În cursul evoluției au putut să apară diferite inserții, deleții sau chiar mutări ale unei secvențe dintr-o zonă cromozomială în alta, astfel încât avem uneori porțiuni care se pretează bine la comparații urmate de porțiuni care nu au nimic în comun. În aceste situații este preferabil a limita comparația numai pentru anumite porțiuni din secvențe – așa numita „**aliniere locală**".

7.3.2. Distanțe

A. Distanța Hamming

Este cea mai simplă exprimare a similarității între două secvențe, fiind dată de:

DH = numărul total de nepotriviri (7.3.2.a)

Se foloseşte pentru alinierea globală.

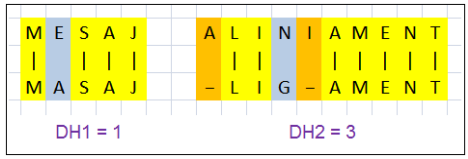


Fig. 7.3.2.a Distanța Hamming

B. Distanța Levenshtein

Se poate îmbunătății evaluarea similarității dacă vom diferenția evenimentele:

- Conservarea o vom cota cu un punctaj pozitiv.
- Substituțiile şi gap-urile le vom cota cu un punctaj negativ, penalizând mai puternic gap-ul decât substituția. Obținem astfel o schemă de scor. Totalul obținut prin aplicarea unei astfel de scheme se numeşte „distanță Levenshtein".

Un model de scor utilizat frecvent este:

$$S = \begin{cases} +2 & pentru\ potrivire\ (conservare) \\ -1 & pentru\ nepotrivire\ (substitutie) \\ -2 & pentru\ gap \end{cases} \quad (7.3.2.b)$$

$$DL = \Sigma Si \quad (7.3.2.c)$$

C. Exemplu:

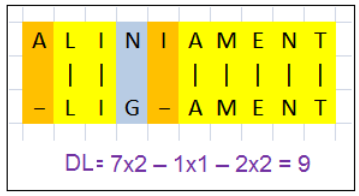


Fig. 7.3.2.b. Distanța Levenshtein

7.3.3. „Problemele" analizei secvențiale

A. Să mai luăm următorul exemplu (figura 7.3.3.).

Am scris normal prima secvență (PARAGINITUL), apoi am scris a doua secvență sub prima literă până aproape de sfârşit, când după R sub I am văzut că, dacă introducem un gap, următoarea literă, U, se va potrivi.

Calculând distanța Levenshtein obținem:

$$DL_3 = +2\,x\,6 - 1\,x\,2 - 2\,x\,4 = +2$$

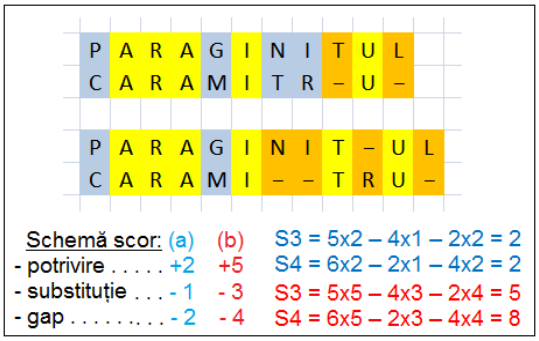


Fig. 7.3.3. Distanța Levenshtein cu 2 scheme de scor.

Observăm însă că, după potrivirea lui I sub I, în a doua secvență avem un T, pe care l-am putea forța, introducând două gap-uri, să se alinieze sub T din prima secvență; mai introducem un gap sus să îl aliniem şi pe U sub U. Obținem deci varianta de dedesubt, cu 3 gapuri în secvența de jos.

Este această aliniere mai bună? Avem într-adevăr mai multe conservări, dar şi mai multe gap-uri.

$$DL_4 = +2\,x\,6 - 1\,x\,2 - 2\,x\,4 = +2$$

Printr-o coincidență, obținem aceeaşi distanță Levenshtein, deci, cele două alinieri ar părea similare.

B. Putem încerca însă şi o altă schemă de scor, coloana (b), folosită destul de frecvent:

$$S = \begin{cases} +5 & conservare \\ -3 & substitutie \\ -4 & gap \end{cases} \qquad (7.3.3.a)$$

Acum avem:

$$DL_3 = +5\,x\,5 - 3\,x\,4 - 4\,x\,2 = +5$$

$$DL_4 = +5\,x\,6 - 3\,x\,2 - 4\,x\,4 = +8$$

Deci cu această schemă, alinierea a doua ar fi mai bună.

C. Concluziile pe care le putem trage din acest exemplu:

i) Rezultatul comparației depinde de schema de scor; avem deci nevoie de scheme de scor cât mai realiste, din care să se înlăture, pe cât posibil, orice arbitrar.

ii) Există mai multe alinieri posibile între secvențe, fiind necesari algoritmi pentru selecția alinierii optime.

În continuare vom studia alegerea alinierii optime în ipoteza unei scheme de scor date, iar în finalul capitolului vom vedea cum putem construi o schemă de scor realistă.

7.4. Programare dinamică – Algoritmul Needleman – Wunsch

7.4.1. Alinierea globală

Aşa cum am văzut mai sus, problema centrală din analiza secvențială este găsirea unui algoritm care să permită selecția alinierii optime între două secvențe, în cadrul unei scheme de scor date.

Primul şi cel mai cunoscut algoritm pentru analiza globală a două secvențe a fost elaborat de Needleman şi Wunsch în 1970. El poate fi aplicat pentru orice fel de secvențe (atât acizi nucleici cât şi proteine).

Vom descrie în continuare algoritmul, pas cu pas, mergând în paralel cu un exemplu.

7.4.2. Principiul algoritmului

A. Plasând cele două secvențe pe cele două axe ale unei matrice, ne propunem să calculăm un element de matrice $F(i, j)$, unde i reprezintă coloana şi j linia.

Considerăm cele două secvențe de comparat:

X, cu elementele X_i, $i = 1, n$

Y, cu elementele Y_j, $j = 1, m$

B. Construim o matrice M x N cu m linii şi n coloane, plasând pe axe – pe axa liniilor secvența X, iar pe cea a coloanelor secvența Y:

X / Y	X_1	X_2	...	X_{i-1}	X_i	...	X_n
Y_1	F(1,1)	F(2,1)	...		F(i,1)	...	F(n,1)
Y_2	F(1,2)	F(2,2)	...			...	
...	...	...	...	...	...	...	...
Y_{i-1}			...	F(i-1, j-1)	F(i, j-1)	...	
Y_i	F(1, j)		...	F(i-1, j)		...	
...	...	...	...	...	...	...	...
Y_m	F(1,m)						F(n,m)

Fig. 7.4.2.a. Matricea elementelor programării dinamice.

C. Valoarea elementului de matrice $F(i,j)$ se calculează după următorul raționament:

- natural este ca elementul $F(i, j)$ să se calculeze din precedentul din secvență, adică din $F(i-1, j-1)$, acordând un „bonus" pentru similaritatea $X_i \equiv Y_j$, sau aplicând o „penalizare" pentru substituția lui cu X_i cu un element $Y_j \neq X_i$.

Valoarea „bonus"-ului sau a „penalizării", notată $S(X_i, Y_j)$ este dată, în general, într-o matrice de substituție". În exemplele anterioare noi am folosit nişte valori generice: +2 (sau +5) pentru conservare, respectiv – 1 (sau – 3) pentru **orice** substituție. Ar fi desigur, preferabil ca toate aceste valori să fie diferite, ținând cont de gradul de asemănare a proprietăților diferiților substituenți; vom scrie în acest caz:

$$F(i,j) = F(i-1, j-1) + S(X_i, Y_j) \qquad (7.4.2.a)$$

- este însă posibil să fie preferabil un „grup" în una din secvențe, pentru a asigura o potrivire mai bună în elementele următoare din secvență. Acest lucru este greu de anticipat, însă va ieşi în evidență după completarea întregii matrice. De aceea, pentru fiecare element de matrice $F(i, j)$, calculăm şi valoarea care ar corespunde în cazul în care într-una din secvențe vom introduce un gap. Astfel, dacă presupunem că am introduce un gap în secvențele Y, am avea:

$$F(i,j) = F(i-1, j) - d \qquad (7.4.2.b)$$

Unde d este „penalizarea" pentru gap; în exemplele anterioare noi am luat valoarea – 2 (respectiv – 4).

- În cazul unui gap în secvența X vom utiliza relația:

$$F(i,j) = F(i, j-1) - d \qquad (7.4.2.c)$$

Schematic putem reprezenta calculul elementului din matrice $F(i,j)$ ca în figura 7.4.2.b.

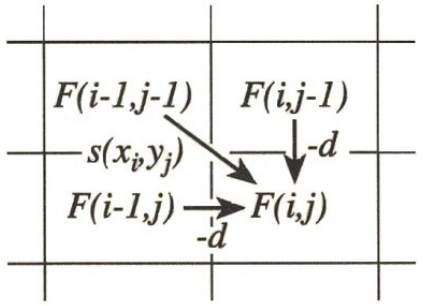


Fig. 7.4.2.b. Algoritmul Needleman-Wunsch.

Algoritmul Needleman – Wunsch pune în poziție centrală alegerea valorii maxime dintre cele trei posibilități prezentate mai sus, exprimată pe relația:

$$F(i,j) = max \begin{cases} F(i-1, j-1) + S(X_i Y_j) \\ F(i-1, j) - d \\ F(i, j-1) - d \end{cases} \qquad (7.4.2.d)$$

7.4.3. Marcarea traseului

A. Pentru a marca varianta aleasă, se obişnuieşte a se plasa în căsuța F (i, j) o săgeată, fie diagonală orientată spre stânga-sus „\", fie la stânga „←", fie în sus „↑", care să indice care a fost varianta utilizată. În cazul în care două variante de calcul dau aceeaşi valoare, vom plasa ambele săgeți.

B. Aplicând aceste reguli, pe rând, la toate elementele matricei vom completa matricea de scor pentru determinarea alinierii optime a celor două secvențe. Algoritmul prin care se realizează „alinierea optimă" se numeşte „trace-back".

C. Se porneşte de la elementul F (n, m), din colțul dreapta-jos al matricei, urmărind săgețile care indică „din care element" a fost calculat. Vom marca astfel fiecare „pas", pornind de la „coada" secvențelor spre început. În cele din urmă vom obține alinierea globală optimă a celor două secvențe.

D. Este posibil ca soluția să nu fie unică şi să avem două (sau chiar mai multe) alinieri, caracterizate prin distanțe identice. Desigur matricea de substituție influențează scorul (valoarea „distanței") pentru fiecare aliniere.

7.4.4. Exemplu

A. Să ilustrăm aplicarea algoritmului Needleman-Wunsch pe un exemplu simplu. Să considerăm secvențele ADN:

X: $C\ G\ T\ A$
Y: $C\ T\ A$

De asemenea vom aplica o matrice de substituție foarte simplă: *potrivire* = + 1, *nepotrivire* = - 1, *gap* = - 2, adică $S\ (i, j) = +\ /\ -1$ şi $d = 2$

B. Construim matricea de aliniere. Plasăm secvența X pe orizontală şi Y pe verticală. Vom „iniția liza" matricea adăugând în plus o linie şi o coloană înainte de începerea fiecărei secvențe, corespunzătoare situației în care primul (primele) element(e) din oricare dintre secvențe ar fi un „gap". Vom nota aceste elemente cu F $(i,0)$ pentru linia inițială, respectiv F (0, j) pentru coloana de inițializare.

	Col	(0)	(1)	(2)	(3)	(4)
Lin			C	G	T	A
(0)		0	-2	-4	-6	-8
(1)	C	-2				
(2)	T	-4				
(3)	A	-6				

Fig. 7.4.4.a. Inițializarea matricei de aliniere

Elementului F (0,0) îi dăm valoare 0, apoi cazul în care gap-urile s-ar succede, pentru fiecare element aplicăm o penalizare de gap, deci elementele liniei de inițializare vor avea valorile: d, 2d, 3d etc. Similar pentru elementele primei coloane. Matricea inițializată în cazul exemplului nostru arată ca în fig. 7.4.4.a.

Calculăm primul element, $F(1,1)$; aplicăm pe rând cele 3 relații din (7.4.2.d) şi obținem:

a: pornind din stânga-sus: $F(1,1) = F(0,0) + S(C,C) = 0 + (+1) = 1$

b: pornind din stânga: $F(1,1) = F(0,1) - d = -2 - 2 = -4$

c: pornind de sus: $F(1,1) = F(1,0) - d = -2 - 2 = -4$

Alegem valoarea maximă (cazul a): $F(1,1) = 1$ şi marcăm săgeata corespunzătoare; rezultatul este vizibil în figura 7.4.4.b.

Continuăm cu elementul $F(2,1)$ pentru care, avem:

a: $F(2,1) = F(1\,0) + S(G,C) = -2 - 1 = -3$

b: $F(2,1) = F(1,1) - d = 1 - 2 = -1$

c: $F(2,1) = F(2,0) - d = -4 - 2 = -6$

Alegem varianta b şi plasăm şi săgeata corespunzătoare.

	Col	(0)	(1)	(2)	(3)	(4)
Lin			C	G	T	A
(0)		0	-2	-4	-6	-8
(1)	C	-2	↖ +1	← -1	← -3	← -5
(2)	T	-4	↑ -1	↖ 0	↖ 0	← -2
(3)	A	-6	↑ -3	↖ ↑ -2	↖ -1	↖ +1

Fig. 7.4.4.b. Completarea matricei de aliniere

Procedăm similar pentru toate elementele de pe linia (1):

$S(T,C): F(3,1) = max(-4-1, -1-2, -6-2) = max(-5, -3, -8)$

$S(A,C): F(4,1) = max(-6-1, -3-2, -8-2) = max(-7, -5, -10)$

Pentru linia a doua:

$S(C,T): F(1,2) = max(-2-1, -4-2, +1-2) = max(-3, -6, -1)$

$S(G,T): F(2,2) = max(+1-1, -1-2, -1-2) = max(0, -3, -3)$

$S(T,T): F(3,2) = max(-1+1, 0-2, -3-2) = max(0, -2, -5)$

$S(A,T): F(4,2) = max(-3+1, 0-2, -5-2) = max(-2, -2, -7)$

În fine pentru ultima linie:

$F(1,3) = max(-4-1, -6-2, -1-2) = max(-5, -8, -3) = -3$

$F(2,3) = max(-1-1, -3-2, 0-2) = max(-2, -5, -2)$

$F(3,3) = max(0-1, -2-2, 0-2) = max(-1, -4, -2)$

$F(4,3) = max(0+1, -1-2, -2-2) = max(+1, -3, -4)$

Matricea de aliniere obținută este prezentată în figura 7.4.4.b

C. Pornim acum algoritmul „trace-back”.

Marcăm căsuța din colțul dreapta-jos $F(4,3)$. Acest element corespunde coincidenței la capătul secvențelor A – A. (figura7.4.4.c).

Observăm că $F(3,4)$ avea o singură săgeată, în diagonală.

Vom alege deci în continuare elementul $F(3,2)$, corespunzător corespondenței T – T. Acest element are de asemenea o singură săgeată, indicând proveniența sa din $F(2,1)$.

	Col	(0)	(1)	(2)	(3)	(4)
Lin			C	G	T	A
(0)		0	-2	-4	-6	-8
(1)	C	-2	↖ +1	← -1	← -3	← -5
(2)	T	-4	↑ -1	↖ 0	↖ 0	← -2
(3)	A	-6	↑ -3	↖ ↑ -2	↖ -1	↖ +1

Fig. 7.4.4.c. Pornirea "trace-back"

Dar acesta are o săgeată spre stânga, adică spre $F(1,1)$ (fig. 7.4.4.d). Cu alte cuvinte, după corespondenta C – C ilustrată de F(1,1), urmează în secvența Y un „gap”, având următoarea căsuță, $F(2,1)$ tot pe orizontală, adică lui „G” din secvența X nu-i corespunde nici un nucleotid în secvența Y.

	Col	(0)	(1)	(2)	(3)	(4)
Lin			C	G	T	A
(0)		0	-2	-4	-6	-8
(1)	C	-2	↖ +1	← -1	← -3	← -5
(2)	T	-4	↑ -1	↖ 0	↖ 0	← -2
(3)	A	-6	↑ -3	↖ ↑ -2	↖ -1	↖ +1

Fig. 7.4.4.d. Rezultatul obținut prin "trace-back"

Rezultatul alinierii îl exprimăm în forma de mai jos (figura 7.4.4.e):

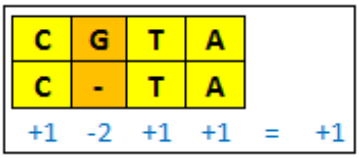


Fig. 7.4.4.e. Alinierea finală

Rezultatul obținut era de așteptat. Observăm că „gap-urile" apar vizibile în matricea de aliniere acolo unde nu se urmează traseul pe diagonală. Dacă apar căsuțe succesive selectate pe aceeaşi linie, atunci gap-ul se va găsi în secvența a doua (cea plasată pe axa verticală a matricii); căsuța cea mai din stânga grupului rămâne în secvență, cealaltă (celelalte, daca sunt mai multe succesiv) vor reprezenta gap-uri. Similar se procedează dacă apar căsuțe succesive pe aceeaşi coloană; în acest caz gap-urile se vor găsi în secvența plasată pe axa orizontală a matricei de aliniere.

7.5. Alinierea locală. Algoritmul Smith – Waterman

7.5.1. Deosebiri față de algoritmul NW

A. Algoritmul Needlemann – Wunsch prezentat mai sus se aplică pentru alinierea globală a două secvențe, adică a secvențelor întregi. Este însă cunoscut faptul că, în evoluția lor, structurile biologice au suferit numeroase modificări, însă au păstrat destul de frecvent anumite porțiuni cu modificări minore, conservând în felul acesta funcționalitatea acestor molecule, în timp ce alte porțiuni din moleculă pot avea evoluții divergente. De aceea, destul de des dorim să detectăm doar domeniile comune din proteine şi să apreciem gradul de similaritate între aceste porțiuni. Acest gen de analiză se numeşte „aliniere locală", care este aplicabilă şi pentru alinierea unor secțiuni extinse de ADN, însă cel mai des se aplică pentru detecția similarității între secvențe divergente, cu origine comună, în care anumite porțiuni se păstrează. Astfel, prin alinierea locală se vor exclude porțiunile terminale din secvențe, porțiuni ce ar fi avut numeroase gap-uri şi un procent înalt de substituții.

B. Pentru alinierea locală, Smith şi Waterman au propus în 1980 un algoritm asemănător cu algoritmul Needlemann – Wunsch, însă introducem următoarele modificări:

- la inițializare, marginile vor avea numai valoarea 0 în loc de penalizările pentru gap-uri;
- în calculul unui element de matrice, vom compara, ca în algoritmul NW valorile obținute din cele trei căsuțe vecine de origine potențială (diagonală stânga-sus, căsuța din stânga, respectiv cea de sus), alegând valoarea maximă; însă dacă toate sunt negative, vom aloca valoarea 0; în felul acesta matricea de aliniere va avea elementele numai pozitive sau 0;
- la "trace-back" vom porni nu din colțul dreapta-jos, ci de la cea mai mare valoare;
- alinierea se poate opri oriunde în matrice, ajungând eventual la marginea superioară sau la cea din stânga în cazul în care vreuna din secvențele aliniate prezintă domeniul de interes chiar la început.

7.5.2. Descrierea algoritmului SW

A. În mod formal putem scrie pentru algoritmul Smith – Waterman următoarele relații și notații:

- H = matricea de aliniere, cu elementele H(i,j)
- m = lungimea secvenței X (pe axa orizontală - sus)
- n = lungimea secvenței Y (pe axa verticală - stânga)
- la inițializare:

$$H(i,0) = 0, 0 \leq i \leq m \quad (i = nr.\ coloanei) \tag{7.5.2.a}$$

$$H(j,0) = 0, 0 \leq j \leq n \quad (j = nr.\ liniei)$$

B. Calculul elementelor matricei H urmează regula:

conservare/substituție

$$H(i,j)\ max \begin{cases} H(i-1,j-1) + s(X_i, Y_j) & conservare/substitutie \\ H(i-1,j) + g(Xi,-) & deletie\ din\ X \\ H(i,j-1) + g(-,Yi) & insertie\ in\ X \\ 0 \end{cases} \tag{7.5.2.b}$$

Uzual se ia penalizarea pentru gap aceeaşi pentru oricare dintre secvențele aliniate X şi Y, adică

$$g(X_i,-) = g(-,Y_i) = -d \tag{7.5.2.c}$$

- valorile $s(X_i,Y_i)$ se iau din matricea de substituție s, la fel ca în cazul alinierii globale; de obicei valorile sunt pozitive pentru conservarea elementului în secvență (potrivire) şi negative pentru substituție.

C. Exemplu

Luăm secvențele:

$$X: GAATTCAGTTA \tag{7.5.2.d}$$

$$Y: CGGATCGA$$

Matricea de substituție s (figura 7.5.2.a) este construită foarte simplu: conservare = +5, substituție = -3, gap = -4 (prescurtat: +5/-3/-4).

	A	C	G	T
A	5	-3	-3	-3
C	-3	5	-3	-3
G	-3	-3	5	-3
T	-3	-3	-3	5
gap = -4				

Fig. 7.5.2.a. Matricea de substituție

Procedăm asemănător cu alinierea globală:

- inițializăm prima linie şi coloană (de indici 0) cu valoarea 0
- calculăm H(1,1) = (0-3,0-4,0-4,0) = 0
- continuăm cu toată linia 1, apoi şi celelalte linii; în cazul în care valoarea aleasă provine din vreuna din căsuțele alăturate, vom marca acest lucru printr-o săgeată ce

indică originea valorii (pot fi şi două săgeți); dacă valorile calculate din căsuțele vecine au fost negative, vom lua valoarea 0 şi nu inserăm nici o săgeată.

În figura 7.5.2.b este prezentată matricea de aliniere după efectuarea acestor calcule.

	Col	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10	(11)
Lin			G	A	A	T	T	C	A	G	T	T	A
(0)		0	0	0	0	0	0	0	0	0	0	0	0
(1)	C	0	0	0	0	0	0	5	← 1	0	0	0	0
(2)	G	0	5	← 1	0	0	0	1	2	6	← 2	0	0
(3)	G	0	5	2	0	0	0	0	0	7	← 3	0	0
(4)	A	0	1	10	7	← 3	0	0	5	3	4	0	5
(5)	T	0	0	6	7	12	← 8	← 4	1	2	8	9	← 5
(6)	C	0	0	2	3	8	9	13	← 9	← 5	4	5	6
(7)	G	0	5	← 1	0	4	5	9	10	14	← 10	← 6	← 2
(8)	A	0	1	10	← 6	← 2	1	5	14	← 10	11	← 7	11

Fig. 7.5.2.b. Matricea de aliniere

- începem procedura "trace-back"; observăm că valoarea maximă din matrice este 14, corespunzătoare elementului H(7,8), care provine în diagonală din H(6,7), legat în sus de H(6,6) şi apoi de H(5,5). De aici sunt două săgeți, deci putem merge fie prin H(4,4) la H(3,4), fie prin H(4,5) şi în diagonală la H(3,4); deci vom avea aici două trasee echivalente; mai departe mergem la H(2,3) şi H(1,2). De fapt şi la pornire, puteam pleca din H(8,7) care avea tot valoarea 14, ajungând în H(6,6), urmând de acolo traseul reprezentat anterior.

Structura alinierilor posibile rămase sunt vizibile în figura 7.5.2.c.

	Col	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10	(11)
Lin			G	A	A	T	T	C	A	G	T	T	A
(0)		0	0	0	0	0	0	0	0	0	0	0	0
(1)	C	0	0	0	0	0	0	5	← 1	0	0	0	0
(2)	G	0	5	← 1	0	0	0	1	2	6	← 2	0	0
(3)	G	0	5	2	0	0	0	0	0	7	← 3	0	0
(4)	A	0	1	10	7	← 3	0	0	5	3	4	0	5
(5)	T	0	0	6	7	12	← 8	← 4	1	2	8	9	← 5
(6)	C	0	0	2	3	8	9	13	← 9	← 5	4	5	6
(7)	G	0	5	← 1	0	4	5	9	10	14	← 10	← 6	← 2
(8)	A	0	1	10	← 6	← 2	1	5	14	← 10	11	← 7	11

Fig. 7.5.2.c. Trace-back pentru alinierea locală

Vom sintetiza rezultatele în figura 7.5.2.d., în care am reprezentat separat variantele de drum din prima porțiune a alinierii, respectiv din a doua parte – în cele din urmă vom avea 4 variante posibile echivalente.

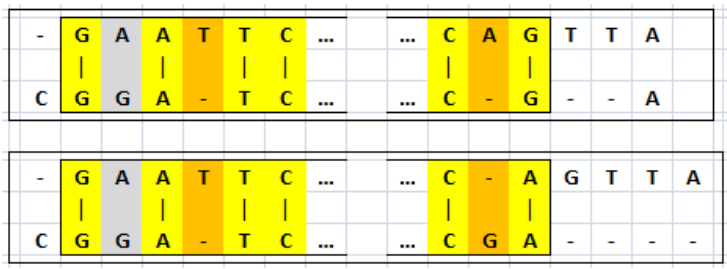


Fig. 7.5.2.d. Alinieri locale posibile

7.6. Modele complexe

7.6.1. Potriviri repetate (Repeated matches)

A. Algoritmul Smith – Waterman găseşte (uzual) o singură cea mai bună aliniere între două secvenţe. Este cunoscut însă faptul că secvenţele reale conţin uneori mai multe copii ale unui domeniu repetat („motiv"). Ar fi deci de dorit să avem un algoritm care să evidenţieze toate aceste regiuni. Există mai multe metode prin care putem realiza acest lucru, una dintre ele, bazată pe „lanţuri Markov ascunse" va fi menţionată în capitolul respectiv. În continuare vom prezenta aici pe scurt o extensie a algoritmului Smith – Waterman, bazat pe programarea dinamică, prin care să putem evidenţia repetările.

B. Să considerăm cele două secvenţe de comparat $X(x_i,i=1,\ldots,n)$ şi $Y(y_j,j=1,\ldots,m)$ şi să presupunem că secvenţa Y conţine un domeniu care se repetă. Să mai reamintim că şi în acest caz, ca şi în cele discutate până acum, porţiunile aliniate conţin nu numai elemente conservate ci şi substituţii sau gap-uri. Intuim deci dificultatea ce apare prin semnalarea oricăror scurte porţiuni din Y ce se aseamănă cu scurte porţiuni din X. De aceea, pentru a obliga algoritmul să semnaleze doar secvenţele cu relevanţă, se introduce un prag, T, (threshold), astfel încât numai regiunile în care scorurile depăşesc pragul vor fi scoase în evidenţă. Pragul se introduce în relaţia de recurenţă pentru iniţializarea primei linii. Astfel, în timp ce la algoritmul de aliniere locală S. W. toate valorile erau zero: H(i,0) = 0, i = 1,…,n, în extensia SW pentru potriviri repetate vom lua H(0,0) = 0, apoi:

$$H(i,0) = max\begin{cases} H(i-1,0) \\ H(i-1,j) - T,\ j = 1,\ldots,m \end{cases} \qquad (7.6.1.a)$$

În felul acesta secvenţa X va fi partiţionată în regiuni care prezintă aliniere cu porţiuni din Y şi regiuni fără potriviri.

Posibila prezenţă a unor valori diferite de zero pe prima linie (linia de iniţializare) permite terminarea unor alinieri când se ajunge la marginea superioară a tabelului prin procedura de trace-back chiar dacă valoarea din căsuţă nu este zero (în timp e în algoritmul clasic SW, alinierea se considera încheiată întotdeauna când se ajungea la o căsuţă cu valoarea zero - indiferent de poziţia ei în matrice).

Vom ilustra detecţia unei potriviri repetate printr-un exemplu.

C. Exemplu

Fie secvenţele proteice X şi Y definite astfel:

X: $DTWGGICDAWHGL$ (7.6.1.b)

Y: $CDAWGEP$

Obţinem alinierea unei porţiuni din Y (zona [C]DAWG) cu două regiuni din X:

D	T	W	G	G	I	C	D	A	W	H	G	L
D	A	W	G	.	.	C	D	A	W	-	G	.

Fig. 7.6.1. Reprezentarea unei alinieri repetate

Am marcat cu un punct (.) regiunile de nepotrivire.

D. Procedura "trace-back" are nişte particularităţi în acest caz. Pentru aplicarea ei procedăm astfel:

- introducem în matricea de aliniere o căsuţă în plus pe prima linie: *H(n+1,0)* cu o valoare calculată după aceeaşi relaţie de recurenţă (7.6.1.a); valoarea din această căsuţă va reprezenta scorul total al tuturor potrivirilor, având valoarea T scăzută pentru fiecare potrivire (deci la k potriviri s-au scăzut kT puncte). Există şi posibilitatea ca valoarea să fie 0, dacă pragul T nu a fost niciodată depăşit.
- din căsuţa *H(n+1,0)* vom trasa o săgeată către căsuţa din care provine (fie cea din stânga ei, fie o altă căsuţă din coloana anterioară); putem deci ca în procedura "trace-back" să avem ,,salturi".
- în continuare procedăm similar cu procedura aplicată în celelalte cazuri.
- de menţionat că, de fiecare dată când ajungem la marginea superioară, dacă valoarea nu este 0 vom avea un salt la căsuţa cu valoarea maximă din coloana din stânga.
- procedura se încheie la o căsuţă cu valoarea 0, oriunde în matrice, cel mai adesea fie pe marginea superioară (deci restul din secvenţa X va fi considerat "gap" terminal), fie pe marginea din stânga (deci "gap-ul" terminal va fi din secvenţa Y).

E. O problemă importantă la detecţia potrivirilor repetate este alegerea pragului T

- o valoare prea mare ar putea exclude unele potriviri, rămânând nedetectate.
- o valoare prea mică divizează mult secvenţele semnalând şi potriviri mai slabe, uneori întâmplătoare (mai ales în secvenţele de acizi nucleici, unde alfabetul secvenţelor are doar 4 litere).

7.6.2. Potriviri suprapuse

A. altă situaţie cu care ne putem întâlni este cazul în care o porţiune mai lungă dintr-o secvenţă are ea însăşi zone repetitive, sau este inclusă în cealaltă secvenţă. Vom fi într-o situaţie asemănătoare cu cea de la potriviri repetate, cu deosebirea că zonele de potrivire din secvenţa de referinţă (notată anterior cu X) nu sunt disjuncte ci se pot suprapune parţial.

Putem întâlni astfel situaţii la compararea fragmentelor genomice de ADN sau la compararea unor secvenţe cromozomale mari.

B. Pentru detecţia acestor acestor situaţii vom face o nouă extensie la algoritmii de programe dinamică , pornind tot de la varianta Smith – Waterman pentru alinierea locală şi vom ţine cont şi de utilitatea pragului T introdus pentru detecţia repetărilor, în cazul în care urmărim şi acest aspect.

Ca elemente specifice ale algoritmului în această extensie avem:
- păstrăm regula alinierii locale de a nu penaliza "gap-urile" terminale (început sau sfârşit de secvenţă)
- marginile se iniţiază cu 0, ca la algoritmul SW; există totuşi versiuni în care se urmăresc şi repetiţiile, în care caz se aplică relaţia (7.6.) cu un prag T ales corespunzător
- pentru calculul elementelor matricei de aliniere se aplică algoritmul Meedlemann – Wunsch al alinierii globale; vom putea avea deci şi elemente negative
- se alege valoarea maximă F_{max} ca valoare maximă de pe linia de jos sau coloana din dreapta (deci nu cea mai mare din matrice)
- procedura "trace-back" va începe de la F_{max} şi va merge până la marginea de sus sau din stânga
- toate celelalte reguli rămân nemodificate.

C. Să luăm din nou un exemplu:

Fie secvenţele X şi Y din figura 7.6.2.

X:	D	A	W	T	L	A	C	E	L	P	A	C	S
Y:		A	C	T	L	A	C	H					
Y:						A	C	T	L	-	A	C	H

Fig. 7.6.2. Alinierea suprapusă a două secvenţe proteice, X şi Y

Se observă că apare o suprapunere între cele două potriviri găsite de algoritmul de aliniere.

7.6.3. Potriviri hibride (Hybrid match conditions)

A. Din cele prezentate anterior rezultă că metodele programării dinamice sunt destul de flexibile putând fi introduse extensii care să evidenţieze elementele pe care dorim să le extragem în procesul de comparaţie a două secvenţe. Fiecare metodă prezentată are avantajele şi dezavantajele sale. De aceea, este bine a avea la dispoziţie proceduri prin care putem scoate în evidenţă anumite particularităţi.

B. Enumerăm în continuare câteva situaţii care necesită abordări specifice, fără a intra în detalii privind aspectele teoretice ale acestor abordări. Pentru detalii privind extensiile algoritmilor în aceste cazuri recomandăm tratatul lui Durbin şi colaboratorii.

Iată câteva exemple care necesită abordări specifice:
- secvenţă repetitivă care tinde să fie găsită în copii tandem neseparate,
- când căutăm secvenţe ce încep la startul ambelor secvenţe dar se pot termina în orice punct,
- când există o probabilitate ridicată ca o secvenţă să fie regăsită integral în cealaltă, dar şi o probabilitate mare se a găsi numai un segment – aşa numitele ,,căutări în familii".

C. Din punct de vedere teoretic cazurile standard sunt limitate, dar putem găsi ceva apropiat. În plus, se pot efectua aşa-numitele ,,post-procesări", adică o rafinare a rezultatului oferit de algoritmi prin selecția variantelor cele mai plauzibile de aliniere, pentru care există alternative justificatoare.

7.7. Gap-uri afine

7.7.1. Baze teoretice

A. Un aspect important, dar neglijat până acum, a fost cel privind penalizarea gap-urilor. Variantele descrise mai sus au aplicat o penalizare uniformă pentru fiecare gap ce era ,,cerut" de algoritm pentru optimizarea scorului. Totuşi, când legăm aspectul formal, exprimat prin relațiile propuse, cu fenomenul real, putem lesne observa că originea gap-urilor, generate de procese de inserție sau deleție în fenomenele de copiere (indiferent de nivel – replicare, transcriere, translație) cuprind cel mai adesea o porțiune dintr-o secvență, ce poate avea lungimi variabile, de la 1 component (nucleotid sau aminoacid) la zeci de componente. Deci ,,gravitatea" fenomenului generator de "gap" în aliniere, este mai curând legată de apariția "per-se" a fenomenului şi mai puțin de dimensiunea sa (adică de numărul de elemente implicate). De aceea ar trebui ă ținem cont de prezența ,,gap-urilor în lanț" şi să ajustăm algoritmul în acest sens.

7.7.2. Tipuri de gap-uri

Din punct de vedere formal vom spune că avem două tipuri principale de penalizare a gap-urilor:

a) penalizare liniară, exprimată prin relația:

$$\gamma(g) = -gd \qquad (7.7.2.a)$$

unde:
- g = nr. gap-uri
- d = penalizarea pentru un grup (este variant utilizată în prezentările anterioare)

b) gap-uri afine, pentru care penalizarea se face conform relației:

$$\gamma(g) = -d - (g - 1) \times e$$

unde:
- „d" este penalizarea pentru primul gap
- „e" este penalizarea pentru un gap din lanț; de obicei e«d.

Pentru a introduce aceste elemente, va fi nevoie să modificăm corespunzător şi relațiile de recurență folosite atât în algoritmii de aliniere globală cât şi în cei de aliniere locală, cu toate extensiile lor.

8. Matrici de substituție

8.1. Introducere

A. În prezentările anterioare ne-am concentrat atenția asupra algoritmilor prin care putem optimiza alinierea a două secvențe. De fiecare dată am acceptat ideea că, în cazul în care se observă conservarea (potrivirea) unui component într-o anumită poziție în cele două secvențe, să tratăm evenimentul ca fapt pozitiv, acordându-i puncte la scorul de aliniere, în timp ce prezența unui gap sau substituția – adică înlocuirea unui element cu altul – să fie penalizate. În acest capitol ne vom ocupa de modul în care stabilim penalizările ce vor fi utilizate în calculul scorului de aliniere, pentru substituția unui element dintr-o secvență, cu un alt element.

B. Ne vom ocupa separat de matricile de substituție utilizate în studiile privind acizii nucleici şi cele din compararea secvențelor proteice.

Există o serie de asemănări fundamentale între abordările utilizate în cele două categorii de molecule, dar şi unele deosebiri. Vom considera aici ca esențiale asemănările, astfel încât nu vom descrie separat elementele constructive pentru fiecare categorie de molecule.

8.2. Matrici de substituție pentru proteine

Este evident că matricile de substituție trebuie să acorde valori în funcție de tipul de substituție . Cea mai simplă variantă ar fi ,,uniformizarea" substituțiilor, variantă prin care toate substituțiile sunt penalizate la fel. Din exemplele prezentate anterior, conform notației formale [+2/-1/-2] sau [+5/-3/-4], valorile din schema de scor au semnificația [conservare/substituție/gap], deci noi am avut exemple în care substituția era penalizată fie cu 1 punct (față de +2 pentru conservare), respectiv cu 3 puncte (față de +5 pentru conservare). Oricum, ne putem imagina că au fost introduse o serie de simplificări, utile din punct de vedere didactic.

Totuşi, scopul de aliniere, care este obiectivul principal în oricare din studiile de aliniere secvențială, depinde în mare măsură de valorile din matricea de substituție. De aceea, ne vom concentra atenția asupra modalităților prin care putem oferi o matrice cu valori rezonabile pentru a fi utilizate în algoritmii de aliniere secvențială (programare dinamică).

8.2.1. Matrici PAM

A. Denumirea PAM provine de la "**P**oint **A**ccepted **M**utations", reprezentând probabilitățile mutațiilor. Să încercăm să explicăm această denumire.

- Ipoteza de bază: fiecare substituție a unui aminoacid (AA) cu alt AA este independentă de alte schimbări anterioare (este clar că ipoteza nu corespunde realității, însă reprezintă o bună aproximație pentru început).

Pentru a găsi nişte valori rezonabile, cât mai aproape de realitate, Margaret Dayhoff (1978) a pornit de la analiza unor date privind substituțiile întâlnite evolutiv între familii

similare de proteine, extrapolându-le pentru distanțe evolutive mari. S-au utilizat numai date confirmate privind secvențele din 71 familii de proteine cu grad de similaritate de cel puțin 85%, totalizând 1572 schimbări; înrudirea secvențelor a fost analizată prin construcția unui arbore filogenetic, folosind metoda parsimoniei (pe care le vom prezenta în capitolul de analiză filogenetică). Prin acest studiu s-a putut s-a putut construi o matrice conținând frecvențele tuturor perechilor de reziduuri între secvențe şi ancestorii lor imediați din arbore. Din această matrice, printr-o serie de operații de normalizare şi scalare în timp s-a obținut o matrice generică (figura 8.2.1.a), numită PAM1 conținând, pentru fiecare aminoacid probabilitatea de a fi substituit de un alt aminoacid într-un anumit interval de timp, când numărul aşteptat de substituții a fost 1% (de unde denumirea PAM1 = Point Accepted Mutation 1%).

Example PAM1 matrix (normalized probabilities multiplied by 10000)

	Ala A	Arg R	Asn N	Asp D	Cys C	Gln Q	Glu E	Gly G	His H	Ile I	Leu L	Lys K	Met M	Phe F	Pro P	Ser S	Thr T	Trp W	Tyr Y	Val V
Ala A	9867	2	9	10	3	8	17	21	2	6	4	2	6	2	22	35	32	0	2	18
Arg R	1	9913	1	0	1	10	0	0	10	3	1	19	4	1	4	6	1	8	0	1
Asn N	4	1	9822	36	0	4	6	6	21	3	1	13	0	1	2	20	9	1	4	1
Asp D	6	0	42	9859	0	6	53	6	4	1	0	3	0	0	1	5	3	0	0	1
Cys C	1	1	0	0	9973	0	0	0	1	1	0	0	0	0	1	5	1	0	3	2
Gln Q	3	9	4	5	0	9876	27	1	23	1	3	6	4	0	6	2	2	0	0	1
Glu E	10	0	7	56	0	35	9865	4	2	3	1	4	1	0	3	4	2	0	1	2
Gly G	21	1	12	11	1	3	7	9935	1	0	1	2	1	1	3	21	3	0	0	5
His H	1	8	18	3	1	20	1	0	9912	0	1	1	0	2	3	1	1	1	4	1
Ile I	2	2	3	1	2	1	2	0	0	9872	9	2	12	7	0	1	7	0	1	33
Leu L	3	1	3	0	0	6	1	1	4	22	9947	2	45	13	3	1	3	4	2	15
Lys K	2	37	25	6	0	12	7	2	2	4	1	9926	20	0	3	8	11	0	1	1
Met M	1	1	0	0	0	2	0	0	0	5	8	4	9874	1	0	1	2	0	0	4
Phe F	1	1	1	0	0	0	0	1	2	8	6	0	4	9946	0	2	1	3	28	0
Pro P	13	5	2	1	1	8	3	2	5	1	2	2	1	1	9926	12	4	0	0	2
Ser S	28	11	34	7	11	4	6	16	2	2	1	7	4	3	17	9840	38	5	2	2
Thr T	22	2	13	4	1	3	2	2	1	11	2	8	6	1	5	32	9871	0	2	9
Trp W	0	2	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	9976	1	0
Tyr Y	1	0	3	0	3	0	1	0	4	1	1	0	0	21	0	1	1	2	9945	1
Val V	13	2	1	1	3	2	2	3	3	57	11	1	17	1	3	2	10	0	2	9901

Fig. 8.2.1.a. Matrice generică PAM1

B. Pentru intervalele mai îndelungate procentul de mutații creşte. Într-o accepțiune a liniarității fenomenului, se poate calcula o matrice pentru un procent n% de mutații acceptate prin ridicarea matricei PAM1 la puterea n:

$$PAM_n = (PAM1)^n \qquad (8.2.1.a)$$

C. Pentru utilizarea matricilor de substituție în analiza secvențială este necesară încă o normalizare, prin care să ținem cont de frecvența cu care apare aminoacidul substituitor în lanțurile proteice.

Să notăm un element din matricea PAM1 cu P(b|a,t=1), reprezentând probabilitatea ca aminoacidul **a** să fie substituit cu **b** într-o unitate de timp arbitrară (t=1) şi cu f_b

probabilitatea de apariție a aminoacidului **b**. Atunci un element din matricea de substituție, S(a,b), reprezentând numărul de puncte ce se atribuie în cazul substituției aminoacidului **a** cu aminoacidul **b** este dat de relația (pentru orice timp t, pe care îl omitem):

$$S(a,b) = k \times \log \frac{P(b|a)}{f_b} \qquad (8.2.1.b)$$

unde k este o constantă.

Observăm că s-a introdus o scară logaritmică, pentru a obține o scară aditivă.

D. Matricile PAM sunt foarte utilizate, fiind posibil a construi variante pentru diferite procente de similaritate. Astfel, pentru un procent de similaritate de 50% se foloseşte matricea PAM80, pentru 40% - PAM120 etc. Cea mai des folosită matrice este matricea PAM250, prezentată în figura 8.2.1.b (Să menționăm că indicele matricei PAM este legat puternic de intervalul de timp în care se acceptă mutația şi nu reprezintă procentul de mutații!)

The PAM250 scoring matrix

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	2																			
R	-2	6																		
N	0	0	2																	
D	0	-1	2	4																
C	-2	-4	-4	-5	4															
Q	0	1	1	2	-5	4														
E	0	-1	1	3	-5	2	4													
G	1	-3	0	1	-3	-1	0	5												
H	-1	2	2	1	-3	3	1	-2	6											
I	-1	-2	-2	-2	-2	-2	-2	-3	-2	5										
L	-2	-3	-3	-4	-6	-2	-3	-4	-2	2	6									
K	-1	3	1	0	-5	1	0	-2	0	-2	-3	5								
M	-1	0	-2	-3	-5	-1	-2	-3	-2	2	4	0	6							
F	-4	-4	-4	-6	-4	-5	-5	-5	-2	1	2	-5	0	9						
P	1	0	-1	-1	-3	0	-1	-1	0	-2	-3	-1	-2	-5	6					
S	1	0	1	0	0	-1	0	1	-1	-1	-3	0	-2	-3	1	3				
T	1	-1	0	0	-2	-1	0	0	-1	0	-2	0	-1	-2	0	1	3			
W	-6	2	-4	-7	-8	-5	-7	-7	-3	-5	-2	-3	-4	0	-6	-2	-5	17		
Y	-3	-4	-2	-4	0	-4	-4	-5	0	-1	-1	-4	-2	7	-5	-3	-3	0	10	
V	0	-2	-2	-2	-2	-2	-2	-1	-2	4	2	-2	2	-1	-1	-1	0	-6	-2	4

Fig. 8.2.1.b. Matricea de substituție PAM250

Se observă că valorile au fost rotunjite la numere întregişi matricea a devenit simetrică. Valorile de pe diagonală reprezintă scorul acordat la conservarea aminoacidului pe poziția respectivă în secvență.

8.2.2. Matrici BLOSUM

A. Matricile PAM au fost construite pornind de la probabilitățile de substituție pentru intervale de timp relativ scurte. S-a constatat că extensia pentru intervale îndelungate (PAM250) se depărtează de realitate. Destul de frecvent ne întâlnim cu substituții succesive în aceeaşi poziție sau cu substituții în bloc, deci o îndepărtare de la ipoteza că fiecare substituție este independentă de alte substituții.

B. Pentru a înlătura aceste neajunsuri, Stephan şi Georgia Henikoff (1992) au propus o nouă variantă de matrice de substituție numită BLOSUM "BLOcks (Amino Acid) SUbstitution Matrix". Ei au analizat 2000 de patternuri de aminoacizi organizați în blocuri, pornind de la bazele de date conținând alinierea multiplă a unor proteine înrudite mai îndepărtat, luând regiunile fără gap-uri şi incluzând în bloc porțiunile (clustere) pentru care procentul de reziduuri identice depăşea un prag L. Calculând frecvențele cu care un reziduu **a** dintr-un cluster se alinia cu un reziduu **b** din alt cluster, s-au putut estima probabilitățile de substituție.

	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	
C	9																				C
S	-1	4																			S
T	-1	1	5																		T
P	-3	-1	-1	7																	P
A	0	1	0	-1	4																A
G	-3	0	-2	-2	0	6															G
N	-3	1	0	-2	-2	0	6														N
D	-3	0	-1	-1	-2	-1	1	6													D
E	-4	0	-1	-1	-1	-2	0	2	5												E
Q	-3	0	-1	-1	-1	-2	0	0	2	5											Q
H	-3	-1	-2	-2	-2	-2	1	-1	0	0	8										H
R	-3	-1	-1	-2	-1	-2	0	-2	0	1	0	5									R
K	-3	0	-1	-1	-1	-2	0	-1	1	1	-1	2	5								K
M	-1	-1	-1	-2	-1	-3	-2	-3	-2	0	-2	-1	-1	5							M
I	-1	-2	-1	-3	-1	-4	-3	-3	-3	-3	-3	-3	-3	1	4						I
L	-1	-2	-1	-3	-1	-4	-3	-4	-3	-2	-3	-2	-2	2	2	4					L
V	-1	-2	0	-2	0	-3	-3	-3	-2	-2	-3	-3	-2	1	3	1	4				V
F	-2	-2	-2	-4	-2	-3	-3	-3	-3	-3	-1	-3	-3	0	0	0	-1	6			F
Y	-2	-2	-2	-3	-2	-3	-2	-3	-2	-1	2	-2	-2	-1	-1	-1	-1	3	7		Y
W	-2	-3	-2	-4	-3	-2	-4	-4	-3	-2	-2	-3	-3	-1	-3	-2	-3	1	2	11	W

Fig. 8.2.2. Matricea de substituție BLOSUM62

C. Din punct de vedere formal, un element al matricei de substituție este dat de relația:

$$S(a,b) = (1/\lambda) \times log[p(a,b)/f_a f_b] \qquad (8.2.2.a)$$

unde λ este o constantă,

$p(a,b)$ – probabilitatea alinierii reziduului a cu b

$f_a f_b$ – probabilitățile (frecvențele) aminoacizilor a şi b

Relația (8.2.2.a) este similară cu (8.2.1.b) ținând cont de faptul că probabilitatea unei substituții a lui a cu b poate fi exprimată prin relația:

$$P(b|a) = p(a,b)/f_a \qquad (8.2.2.b)$$

D. Matricile BLOSUM au valori destul de apropiate de matricile PAM echivalente, deşi au pornit de la baze de date diferite şi au fost construite în mod diferit.

În practică se folosesc uzual matricile BLOSUM62 şi BLOSUM50. Varianta BLOSUM50, cu procent de similaritate de 50% corespunde unui timp evoluționar mai lung, deci este mai potrivită pentru secvențele mai divergente, conținând eventual şi

gap-uri, în timp ce BLOSUM62, care este folosită şi mai frecvent, ete considerată standard pentru porțiuni fără gap-uri.

În figura 8.2.2 este prezentată matricea BLOSUM62.

8.3. Matrici de substituție pentru acizi nucleici

La fel ca şi în cazul proteinelor, punctul de plecare pentru analiza substituțiilor porneşte de la analiza evolutivă a structurilor moleculare din materia vie.

Matricile de substituție pentru acizii nucleici s-au construit înaintea celor pentru proteine, pornind chiar de la o abordare teoretică a modelelor evolutive.

8.3.1. Matricea Jukes – Cantor

A. Cea mai simplă versiune a unei matrici de substituție a fost abordată de Jukes şi Cantor (1969). În modelul lor ei au considerat că:

- fiecare site dintr-o frecvență polinucleotidică poate fi tratat ca independent (adică substituția într-o poziție este independentă de alte substituții),
- toate substituțiile au aceeaşi probabilitate,
- nu sunt luate în considerare inserții au deleții,
- toate nucleotidele au aceeaşi probabilitate de apariție într-o frecvență,
- probabilitatea unei substituții este constantă în timp.

B. Din punct de vedere formal, dacă notăm cu α probabilitatea unei substituții într-un interval de timp **t** (numită rata substituției), atunci matricea probabilităților acestor substituții vor fi cele din figura 8.3.1.a, unde A,C,G,T sunt simbolurile celor 4 tipuri de nucleotide ale ADN.

	A	C	G	T
A	1 - 3α	α	α	α
C	α	1 - 3α	α	α
G	α	α	1 - 3α	α
T	α	α	α	1 - 3α

Fig. 8.3.1.a. Matricea probabilităților substituțiilor în modelul Jukes – Cantor

C. Dacă am lua probabilitatea unei substituții de 1% am obține matricea generică PAM1 pentru ADN conform modelului Jukes – Cantor prezentată în figura 8.3.1.b.

	A	C	G	T
A	0.99			
C	0.0033	0.99		
G	0.0033	0.0033	0.99	
T	0.0033	0.0033	0.0033	0.99

Fig. 8.3.1.b. Matricea generică PAM1 în modelul Jukes – Cantor

D. Aplicând şi în acest caz o relație logaritmică, cu logaritm în baza 2, obținem matricea de substituție PAM1 prezentată în figura 8.3.1.c.

	A	C	G	T
A	2			
C	-6	2		
G	-6	-6	2	
T	-6	-6	-6	2

Fig. 8.3.1.c. Matricea de substituție PAM1 pentru ADN în modelul Jukes – Cantor

S-au luat valori întregi (s-a aproximat $1/300 \approx 2^{-8}$), şi probabilitățile la echilibru $p_A = p_C = p_G = p_T = ¼$.

8.3.2. *Matricea Kimura*

A. Condițiile simplificatoare ale modelului Jukes – Cantor au permis o abordare mai facilă a substituțiilor în acizii nucleici, dar cu riscul unor aproximări mai grosolane.

O primă îmbunătățire a fost realizată de Kimura (1980) care a luat în considerare marea diferență între probabilitățile de tranziție şi cele de transversie. (Reamintim aici că tranzițiile reprezintă substituții în cadrul aceleiaşi clase – purinică sau pirimidinică, adică substituții între A şi G, respectiv între C şi T, iar transversiile sunt substituții între clase, adică A sau G cu C sau T). Probabilitățile transversiilor sunt mai mici decât ale tranzițiilor, fiind vorba de substituții între molecule cu configurație, dimensiune şi proprietăți diferite.

O schemă a acestor substituții, împreună cu probabilitățile lor în unitatea de timp este prezentată în figura 8.3.2.a.

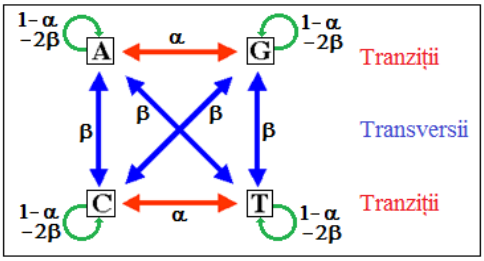


Fig. 8.3.2.a. Tranziții şi transversii

B. Din punct de vedere formal matricea probabilităților tuturor substituțiilor posibile se scrie asemănător ca mai sus. Vom nota cu α rata tranzițiilor şi cu β rata transversiilor (β«α). Atunci matricea probabilităților substituțiilor conform modelului Kimura arată ca cea din figura 8.3.2.b.

	A	C	G	T
A	1 - α - 2β	β	α	β
C	β	1 - α - 2β	β	α
G	α	β	1 - α - 2β	β
T	β	α	β	1 - α - 2β

Fig. 8.3.2.b. Matricea probabilităților substituțiilor în modelul Kimura

C. Putem şi în acest caz să construim o matrice PAM1. Vom lua între α şi β raportul $\alpha = 3\beta$. În acest caz matricea generică PAM1 va fi cea din figura 8.3.2.c.

	A	C	G	T
A	2			
C	-6	2		
G	-6	-6	2	
T	-6	-6	-6	2

Fig. 8.3.2.c. Matricea generică PAM1 pentru ADN conform modelului Kimura

D. Matricea *log odds* corespunzătoare celei din figura 8.3.2.c. este cea din figura 8.3.2.d., aplicând din nou logaritmii în baza 2 şi rotunjind la valori întregi.

	A	C	G	T
A	2			
C	-7	2		
G	-5	-7	2	
T	-7	-5	-7	2

Fig. 8.3.2.d. Matricea de substituție PAM1 pentru acizi nucleici conform modelului Kimura

8.4. Testarea semnificației alinierii

8.4.1. Scoruri

A. După cum am observat din abordările anterioare, abordarea probabilistică ocupă o poziție importantă în studiile de bioinformatică. De aceea, va trebui să luăm în considerare şi faptul că anumite alinieri între frecvențe au o anumită probabilitate de a apărea absolut din întâmplate. Dorim deci să facem o evaluare a acestei probabilități pentru a oferi un anumit nivel de confidență concluziilor noastre.

B. Se poate demonstra că, în cazul în care încercăm să aliniem două secvențe, X şi Y, de lungime **n**, şi respectiv **m**, şi acordăm tuturor alinierilor posibile un scor, atunci numărul E al alinierilor cu un scor minim S este dat de relația:

$$E = k \times n \times m \times e^{-\lambda \times S} \qquad (8.4.1.a)$$

unde k şi λ sunt constante, fiind parametrii statistici cu scorul S. Putem face şi o reprezentare grafică, luând

$$S = 10 \times logx \qquad (8.4.1.b)$$

şi vom obține o distribuție Poisson, reprezentată în graficul din figura 8.4.1.

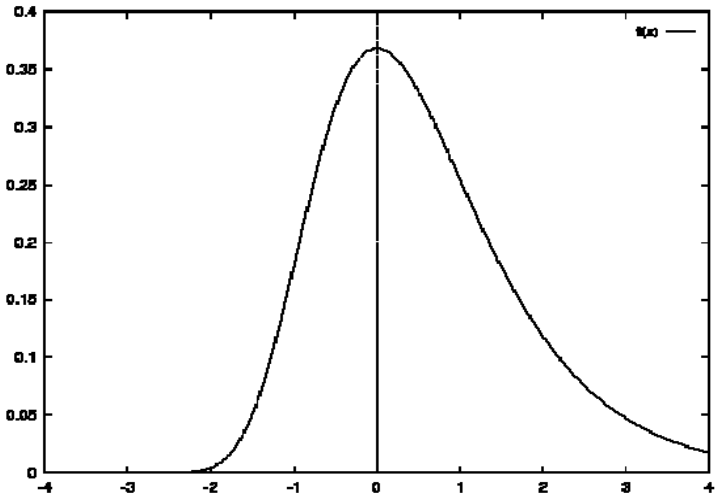


Fig. 8.4.1. Distribuția scorurilor normalizate după lungime

8.4.2. Semnificația scorurilor

A. Pentru semnificația alinierii să reamintim că pentru exprimarea cantității de informație dintr-un scor folosim ca unități de măsură:
- bit-ul, când se lucrează cu logaritmi în baza 2
- nat-ul, când se lucrează cu logaritmi naturali.

B. Conversia la biți se face astfel:

$$S' = (\lambda S - lnk)/ln2 \qquad (8.4.2.a)$$

şi atunci vom avea pentru E relația:

$$E = n \times m \times 2^{-S'} \qquad (8.4.2.b)$$

C. Exprimăm acum valoarea probabilității de a obține un scor minim S la întâmplare:

$$P = 1 - e^{E} \qquad (8.4.2.c)$$

D. Deseori nici nu mai calculăm probabilitatea din relația (8.4.2.c.), ci doar impunem condiția ca exponentul să fie cât mai mic (E«1) adică E să fie semnificativ mai mic decât unitatea. Prin logaritmarea relației (8.4.1.a.) putem scrie condiția

$$S > T + \frac{log(m\,n)}{\lambda} \qquad (8.4.2.d.)$$

T fiind o constantă dependentă de k.

Valorile care se obțin în mod curent pentru P din relația (8.4.2.c.) sunt foarte mici, de ordinul 2^{-10} până la 2^{-100}. Pentru valori P > 0.1% nu se mai consideră asemănări între secvențe, eventualele coincidențe se vor atribui întâmplării.

9. Alinierea multiplă

Analiza secvențială cuprinde, gradat, analiza secvențelor individule, compararea a două secvențe sau compararea mai multor secvențe, numită „aliniere multiplă”, prescurtat MSA (Multiple Sequence Alignment).

9.1. Semnificația alinierii multiple

Pornind de la datele experimentale biologii au izbutit să realizeze manual alinierea de bună calitate a mai multor secvențe, bazându-se pe o serie de cunoştințe acumulate din experiență.

9.1.1. Factorii luați în considerare pentru MSA

Biologii includ pe lista factorilor importanți în alinierea, multiplă atât cunoştințe privind evoluția secvențelor proteice cât şi elemente structurale. Să trecem în revistă aceşti factori:

- alinieri specifice „pe coloane” ale unor componente ce prezintă un înalt grad de conservare sau reziduuri hidrofobre mascate,
- influența structurii secundare şi terțiare, cum ar fi alternanța componentelor hidrofile şi hidrofobe pe coloanele fâşiilor beta,
- pattern-uri de inserții şi deleții care tind să alterneze cu blocuri de secvențe conservate,
- constrângeri impuse de relații filogenetice.

9.1.2. Obiective în alinierea multiplă

A. Dezvoltarea metodelor automate pentru alinierea multiplă ocupă o poziție centrală în bioinformatică. În principiu ele îşi propun să atribuie alinierii un scor care să reflecte calitatea alinierii. Totuşi trebuie să facem distincție între alinierea „optimă” după un anumit criteriu de scor şi alinierea cea mai bună ținând cont şi de criterii structurale şi de evoluție. De aceea se insistă actualmente pe modalitățile de a transpune în algoritmii de scor criteriile biologice.

B. În principiu, în MSA scriem secvențele una sub alta aliniind reziduurile „omoloage” în coloane. Termenul „omolog” (*homologous*) este luat atât în sens structural cât şi de evoluție. Se apreciază că elementele plasate pe o coloană ocupă poziții similare în structura tridimensională a moleculelor şi toate provin cel mai probabil de la un ancestor comun. Identificarea structurală şi de evoluție a pozițiilor omoloage este dificilă şi adesea găsim mai multe alinieri potențial corecte (deci avem unele ambiguități), datorită imposibilității ca două secvențe cu unele diferențe să fie superpozabile în întregime. Unul dintre cele mai bune exemple, în care structura se conservă foarte bine chiar dacă apar diferențe secvențiale este oferit de „familia globinelor”.

C. Chiar dacă – în principiu – admitem că există o singură aliniere corectă evolutiv, în practică este chiar mai dificilă inferența evolutivă decât cea structurală. Aceasta datorită faptului că, pentru alinierea structurală avem criterii clare (cum ar fi

coordonatele spațiale din structura tridimensională determinate prin studii cristalografice sau RMN), în timp ce pentru alinierea evolutivă istoria unei familii de secvențe nu se determină prin metode independente. În plus, secvențele diverg mai rapid decât structurile.

D. În cazul alinierii secvențelor foarte asemănătoare, adesea se poate găsi o aliniere neambiguă, însă, în general, pentru cazurile de interes trebuie să reținem că nu există vreun mijloc obiectiv pentru a defini o aliniere corectă neambiguă. Adesea avem în familiile analizate proteine ce au în comun doar vreo 30% secvențe cu alinieri locale în perechi de bună calitate.

Ne ajută însă deseori câte un mic subset de reziduuri cheie identificabile, ce pot fi aliniate fără ambiguități.

E. Testarea calității alinierii trebuie să țină cont de aceste elemente. În loc să forțăm algoritmii să producă alinieri identice cu cele ce pot fi obținute manual, atenția pe subseturile din coloanele ce corespund reziduurilor cheie.

9.2. Scop şi motivație pentru alinierea multiplă

Vom prezenta în continuare sintetic punerea problemei în alinierea multiplă şi o serie de aplicații în care MSA şi-a dovedit utilitatea.

9.2.1. Punerea problemei

A. Fiind dat un set de mai multe secvențe şi o metodă se scor pentru aliniere, să se determine corespondențele între secvențe astfel încât scorul de aliniere să fie maxim.

B. Revenim aici la comentariul anterior privind criteriile incluse în scor; avem deseori posibilitatea de a aborda aceleaşi secvențe prin mai multe scheme de scor.

9.2.2. Motivație

Rezultatele MSA pot fi folosite pentru:

- stabilirea datelor de intrare pentru analiza filogenetică,
- determinarea istoriei evolutive a unui set de secvențe, precizând în ce punct au apărut anumite mutații,
- detecția unor „motive” comune într-un set de secvențe (de exemplu, secvențe de AND care leagă aceeaşi proteină),
- caracterizarea unui set de secvențe (de exemplu o familie de proteine),
- construcția profilelor pentru căutarea bazelor de date de secvențe (de exemplu, Psl-BLAST).

9.3. Scoruri pentru alinierea multiplă

9.3.1. Privire generală

A. Algoritmii pentru stabilirea unor scoruri de MSA pornesc de la o serie de ipoteze simplificatoare. O ipoteză folosită în majoritatea metodelor este independența coloanelor individuale.

B. Formula generală a unui scor MSA este:

$$Score(m) = G + \sum S(m_i) \qquad (9.3.1.a)$$

unde: G este o funcție de gap, iar i este indicele care precizează coloana, deci $S(m_i)$ este un scor al coloanei „i".

Dintre metodele uzuale ne vom opri pe scurt la:
- suma perechilor,
- entropia minimă.

9.3.2. Suma perechilor

Prin această metodă se calculează suma scorurilor din alinierea perechilor, folosind relația:

$$S(m_i) = \sum_{k<l} S\left(m_i^k, m_i^l\right) \qquad (9.3.2.a)$$

unde am notat cu m^k_i caracterul (simbolul) din secvența „k" în coloana „i". Elementele s sunt elementele matricei de substituție.

9.3.3. Entropia minimă

A. În această metodă ideea de bază este încercarea de minimizare a entropiei informaționale a fiecărei coloane. Sunt „bune" coloanele ce pot fi comunicate cu puțini biți (care au de fapt o bună conservare a unui element).

Reamintim, din teoria informației, faptul că un cod optim foloseşte un număr de $-\log_2 p$ biți pentru a codifica un mesaj/simbol cu probabilitatea p.

B. În MSA, mesajul este considerat pe coloană. Entropia informațională a unei coloane va fi:

$$S(m_i) = -\sum_a c_{ia} \times \log_2 p_{ia} \qquad (9.3.3.a)$$

în care:
- m_i = coloana „i" din secvența/alinierea „m"
- c_{ia} = de câte ori apare caracterul „a" în coloana „i"
- p_{ia} = probabilitatea caracterului „a" în coloana „i".

9.3.4. Reprezentări grafice

A. Pentru alinierea multiplă se foloseşte uzual o reprezentare intuitivă: plasarea secvențelor una sub alta, astfel încât să fie vizibile coincidențele, substituțiile şi gap-urile. Deseori se marchează cu un chenar sau culori diferite coloanele importante (cu conservare puternică) (figura 9.3.4.a.).

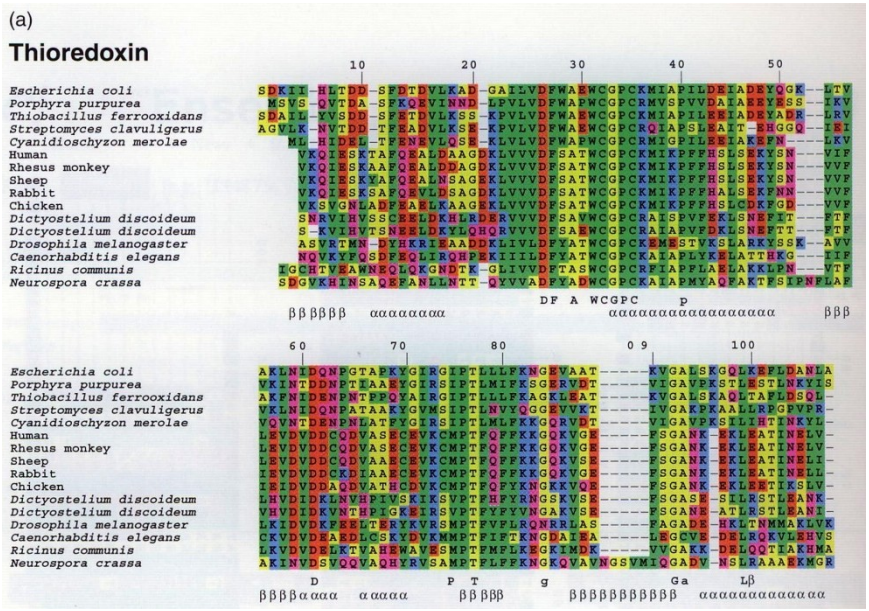


Fig. 9.3.4.a. Reprezentarea alinierii multiple cu coloane

B. În practică se mai foloseşte şi o prezentare sugestivă, în care pentru fiecare poziție din secvență se scriu deasupra simbolurile găsite în aceea poziție cu litere de diverse dimensiuni - cu cât simbolul respectiv a fost mai frecvent în acea coloană, cu atât litera este mai mare. În plus, literele pot avea şi culori diferite (în cazul aminoacizilor) reprezentând caracterul polar/nepolar, acid/bazic. O astfel de reprezentare este redată în figura 9.3.4.b.

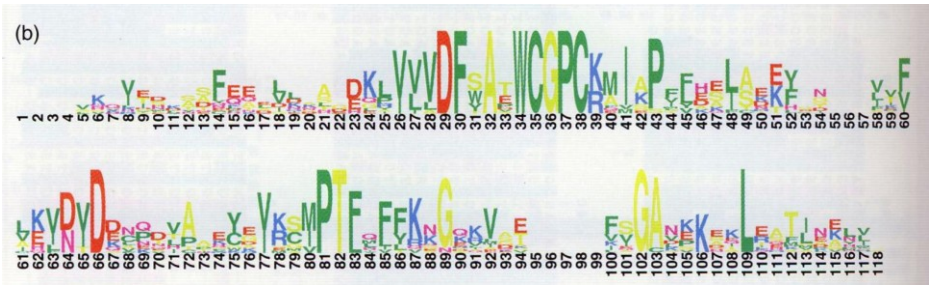


Fig. 9.3.4.b. Reprezentarea alinierii multiple cu simboluri

9.4. *Algoritmi pentru alinierea multiplă*

S-a dezvoltat o largă varietate de algoritmi pentru MSA, pornind adesea de la algoritmii dezvoltați pentru compararea a două secvențe.

Vom trece în revistă cei mai cunoscuți şi utilizați algoritmi.

9.4.1. *Programare dinamică*

Succesul programării dinamice în cazul comparării a două secvențe a făcut ca algoritmii să fie generalizați pentru SA.

În cazul comparării a k secvenţe ar fi necesară construcţia unei matrici de dimensiune k, în care fiecare element să reprezinte scorul pentru k secvenţe în loc de două. Numărul de combinaţii cu gap în oricare secvenţă (exceptând gap în toate) creşte rapid cu k, fiind 2^k-1. Observăm deci că programarea dinamică devine complet nepractică.

9.4.2. Metode de aliniere progresivă

A. Principiul alinierii progresive

O abordare simplă a alinierii multiple este oferită de alinierea progresivă. Aceasta se construieşte printr-o succesiune de alinieri pereche (compararea a două secvenţe). Se iau două secvenţe, se aliniază printr-una din metodele clasice (cel mai adesea prin programare dinamică, cu algoritmul Needleman – Wunsch, Smith – Waterman sau alte variante) din care se obţine o nouă secvenţă care le sintetizează. Se ia apoi a treia secvenţă, care se compară cu sinteza primelor două obţinând o nouă aliniere/secvenţă sintetică a celor 3. Procesul urmează iterativ până la epuizarea celor k secvenţe.

B. Variante de aliniere progresivă

Au fost propuse mai multe variante de aliniere progresivă în funcţie de:

- ordinea în care sunt introduse secvenţele la aliniere, element care poate influenţa puternic rezultatul final,
- maniera de parcurgere a progresiei – adăugarea liniară a câte unei secvenţe sau gruparea secvenţelor întâi pe familii şi reunirea ulterioară,
- procedurile de scor ale alinierilor.

C. Caracterul euristic al alinierii progresive

Este evident faptul că în abordarea descrisă aici, procedura de optimizare nu este separată de calcularea scorului, neexistând o optimizare a unui scor global. Totuşi, alinierea progresivă este rapidă, simplă şi, dacă se urmează o procedură de ierarhizare în succesiunea secvenţelor la aliniere, rezultatele sunt rezonabile.

Se recomandă alinierea la început a perechilor celor mai apropiate, existând şi aici mai multe variante, care vor fi prezentate sumar.

9.4.3. Algoritmul MSA

Vom prezenta pe scurt un algoritm de construcţie a unei secvenţe sintetice. Vom considera cazul unor secvenţe de acizi nucleici.

A. Reprezentarea unei secvenţe prin matricea coeficienţilor

Fie o secvenţă $X(x_1x_2...x_i...x_n)$. Elementele din secvenţă au valori dintr-un alfabet al tipului de secvenţă. Pentru a acoperi cazul general din analiza secvenţial, când într-o anumită poziţie putem avea un gap, alfabetul are m+1 valori posibile, adică în cazul acizilor nucleici avem 5 valori posibile (a, c, g, t, -), iar în cazul proteinelor avem 21 valori posibile (cu „-" am notat un gap).

Vom nota vectorul valorilor posibile cu V. În cazul acizilor nucleici vom avea:

$$V(j) = \{a, c, g, t, \text{“-”}\} \qquad (9.4.3.a)$$

adică $v_1 = V(1) = a$, $v_2 = V(2) = c$, etc.

Pentru orice secvență primară putem construi o matrice a coeficienților T astfel:

$$Tx(i,j) = \begin{cases} 1 \; daca \; x_i = v_j \\ 0 \; daca \; x_i \neq v_j \end{cases} \qquad (9.4.3.b)$$

De exemplu, pentru secvența ***X = (cgga-tg),*** matricea T(X) va avea valorile din tabelul (9.4.3.a)

Tabelul 9.4.3.a. Matricea coeficienților pentru secvența X

	c	g	g	a	-	t	g
a	0	0	0	1	0	0	0
c	1	0	0	0	0	0	0
g	0	1	1	0	0	0	1
t	0	0	0	0	0	1	0
-	0	0	0	0	1	0	0

B. Sinteza a două secvențe

Fie secvențele X şi Y aliniate printr-unul din algoritmii clasici pentru perechi. Dacă nu au lungimi egale, vom completa secvența mai scurtă cu gap-uri.

Deci putem nota $X(x_1x_2...x_i...x_n)$ şi $Y(y_1y_2...y_i...y_n)$. Construim matricea T_z a coeficienților secvenței sintetice z ce va rezulta în urma combinării X cu Y.

Calculăm elementele matricei T_z cu relațiile:

$$Tz(i,j) = [T_x(i,j) + T_y(i,j)]/2 \qquad (9.4.3.c)$$

Să completăm exemplul început mai sus al secvenței ***X = (cgga-tg)***, cu ***Y = (tga- -tt).*** Matricea coeficienților va fi (tabelul 9.4.3.b):

Tabelul 9.4.3.b Matricea coeficienților sintezei Z = (XY)

X Y	c t	g g	g a	a -	- -	t t	g t
a	0	0	0.5	0.5	0	0	0
c	0.5	0	0	0	0	0	0
g	0	1	0.5	0	0	0	0.5
t	0.5	0	0	0	0	1	0.5
-	0	0	0	0.5	1	0	0

C. Sinteza a k secvențe

Relația (9.4.3.c) se va generaliza devenind:

$$T_z(i,j) = [\sum_k T_k(i,j)]/k \qquad (9.4.3.d)$$

D. Matricea de substituție

Față de matricile de substituție clasice, în cazul alinierii multiple vom introduce şi gap-ul ca unul din simbolurile posibile. Să mai facem şi observația că vom avea şi situația în care un gap se conservă peste mai multe secvențe din set, având o penalizare mai mică decât un grup nou. Un model de matrice utilizabilă în alinierea multiplă este redat în tabelul 9.4.3.c.

Tabelul 9.4.3.c. Model de matrice de substituție pentru alinierea multiplă a secvențelor de ADN

	a	c	g	t	-
a	4	-2	-1	-2	-3
c	-2	4	-2	-1	-3
g	-1	-2	4	-2	-3
t	-2	-1	-2	4	-3
-	-3	-3	-3	-3	1

E. Programarea dinamică cu secvențe sintetice

Procedura de programare dinamică poate fi aplicată în cazul în care una sau amblele secvențe sunt sintetice, deosebirea că fiecare poziție din secvență intră în program doar cu coeficientul său de pondere.

9.5. Modele de ordonare

În metodele euristice de aliniere multiplă, cum este alinierea progresivă, ordinea în care sunt luate secvențele devine foarte importantă. În acest sens s-au propus mai multe variante. Vom prezenta în continuare două dintre ele, cel mai des folosite.

9.5.1. Modelul „stea”

A. Se dau k secvențe ce urmează a fi aliniate: $x_1,\ldots, x_k$.

– Se alege o secvență x_c ca şi „centru”.
– Pentru fiecare secvență $x_i \neq x_c$ se determină între x_i şi x_c o aliniere optimală.
– Se reunesc alinierile perechii.
– Rezultatul de aliniere multiplă se obține prin agregarea alinierilor perechii.

B. Pentru alegerea centrului se încearcă fiecare secvență ca centru şi se ia cea mai bună aliniere multiplă rezultată. Procedura este laborioasă însă oferă o alegere rezonabilă a centrului.

C. Aplicarea modelului stea este ilustrată în figurile 9.5.1.a şi 9.5.1.b

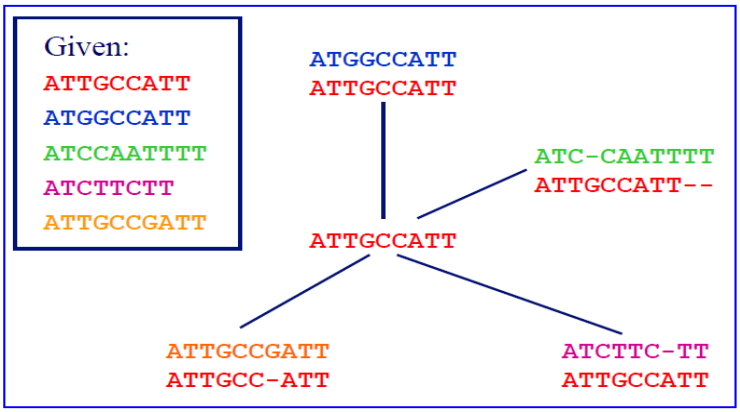


Fig. 9.5.1.a. Aplicarea modelului “stea” pentru nun set de 5 secvențe

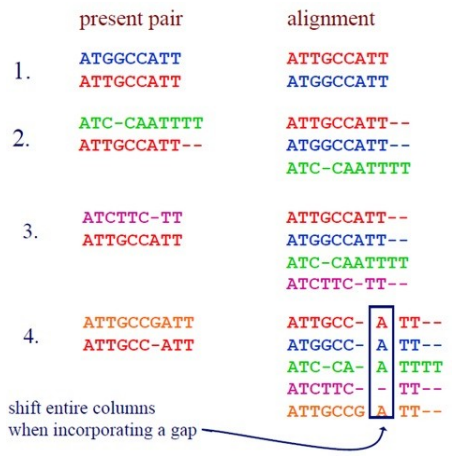


Fig. 9.5.1.b. Introducerea unui gap în modelul „stea"

9.5.2. Modelul "arbore"

Varianta care stă la baza unui soft folosit frecvent este modelul „arbore".

A. Ideea de bază este organizarea alinierii folosind un „arbore ghid" (guide tree), în care:

– frunzele reprezintă frecvențele,
– nodurile interne reprezintă alinieri.

(Noțiunile fundamentale despre arbori vor fi introduse în capitolul de analiză filogenetică).

B. Alinierile de determină pornind de la bază în sus, iar alinierea multiplă rezultată la rădăcina arborelui.

C. Varianta uzuală este aplicată în programul CLUSTAL W (Thompson, 1994).

În funcție de nodul intern din arbore, putem avea de aliniat:

– o secvență cu o secvență
– o secvență cu un profil (aliniere parțială)
– un profil cu un profil.

Pentru cazul profilelor se recomandă scorul SP.

În figura 9.5.2 este prezentat un model de arbore folosit pentru alinierea multiplă.

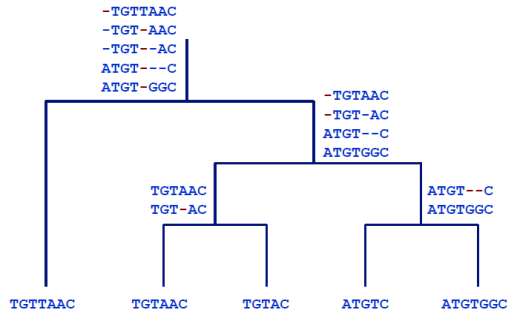


Fig. 9.5.2. Modelul „arbore" construit pentru alinierea multiplă a 5 frecvențe

10. Lanțuri Markov

În abordările anterioare am considerat că probabilitatea de a găsi un anumit element/reziduu (aminoacid în cazul unei secvențe de acid nucleic), este independentă de elementele ce ocupă pozițiile învecinate. Există însă numeroase date experimentale care arată că întâlnim adesea regiuni în care apar cu predilecție anumite perechi sau chiar microsecvențe. În cele ce urmează vom aborda analiza secvențială în acest context.

10.1. Lanțuri Markov simple

10.1.1. Descrierea unui lanț Markov

A. Pentru a lua în considerare dependența probabilității de a găsi un element într-o secvență de elementul anterior, vom construi o schemă aplicabilă în cazul secvențelor de acizi nucleici, în care alfabetul nostru este format din numai 4 litere: A, C, G şi T. (figura 10.1.1)

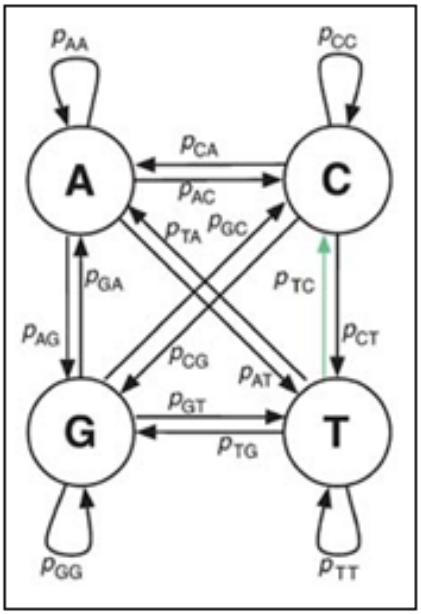


Fig. 10.1.1. Probabilitatea de succesiune în acizii nucleici

B. Am notat indicii probabilităților în ordinea în care apar elementele în secvență, de exemplu p_{AC} este probabilitatea ca C să fie precedat de A, cu alte cuvinte probabilitatea tranziției A→C. Să mai menționăm aici că termenul „tranziție” în analiza secvențelor individuale, cum este cazul aici, are simpla semnificație ca un element să (de ex.: A) să fie urmat de un alt element (de ex.: C) şi nu semnificația din matricile de substituție!

10.1.2. Probabilitatea unei secvențe

A. Din punct de vedere formal putem nota probabilitatea unei tranziții de la elementul „s" la „t" astfel:

$$a_{st} = P(x_i = t|x_{i-1} = s) = P(x_t|x_s) \quad (10.1.2.a)$$

Dacă notăm $P(x_s)$ probabilitatea de a găsi elementul „s", atunci, probabilitatea de a găi perechea „st":

$$P(x_t,x_s) = P(x_t|x_s) \times P(x_s) \quad (10.1.2.b)$$

B. Proprietatea fundamentală a unui lanț Markov este că probabilitatea fiecărui element x_i depinde numai de simbolul precedent $x_{i-1,}$ nu de întreaga secvență din față, adică:

$$P(x_i|x_{i-1,}\ldots,x_1) = P(x_i|x_{i-1} = a_{xi\text{-}1x2} \quad (10.1.2.c)$$

Putem aplica recurent (10.1.2.c) pentru o frecvență de lungime L, notată:

$X = [x_1x_2\ldots x_L]$ și obținem:

$$P(X) = P(x_L|x_{L-1}) \times P(x_{L-1}|\, x_{L-2})\ldots P(x_2|x_1) \times P(x_1) \quad (10.1.2.d)$$

$$P_{(x)} = P_{(x_1)} \quad \prod_{i=2}^{L} a_{x_{i-1}x_i} P(x_{i|}x_{i-1}) \quad (10.1.2.e)$$

Deci probabilitatea unei secvențe se calculează din produsul tuturor tranzițiilor de la probabilitatea primului simbol din frecvență.

C. Pentru omogenizarea relației (10.1.2.e.), adică introducerea primului simbol tot sub forma unei „tranziții", s-a convenit completarea schemei din figura 10.1.1 cu încă o stare denumită „Început" (sau "Begin") din care pot exista tranziții către oricare din simbolurile ce pot fi întâlnite în frecvență (figura 10.1.2). Observăm că starea „B" de început nu are decât săgeți emergente. În acest context vom putea nota probabilitatea ca secvența X să înceapă cu un anumit simbol, de exemplu „s", ca pe o tranziție din starea „B" în „s":

$$P(x_1 = s|B) = a_{Bs} \quad (10.1.2.f)$$

Relațiile (10.1.2.d.) și (10.1.2.e) devin:

$$P(X) = P(x_L|x_{L-1} \times P(x_{L-1}|x_{L-2}) \ldots P(x_2|x_1) \times P(x_1|B) \quad (10.1.2.d')$$

$$P(X) = \prod_{i=1}^{L} a_{x_{i-1}x_i} \quad (10.1.2.e')$$

Putem proceda similar și pentru încheierea secvenței, introducând încă o stare specială, denumită „Sfârșit" (sau "End"), care va avea numai săgeți incidente. De exemplu, dacă secvența X se va termina cu simbolul „t" vom putea marca acest lucru ca o tranziție din starea „t" în „E":

$$P(E|x_L = t) = a_{tE} \quad (10.1.2.g)$$

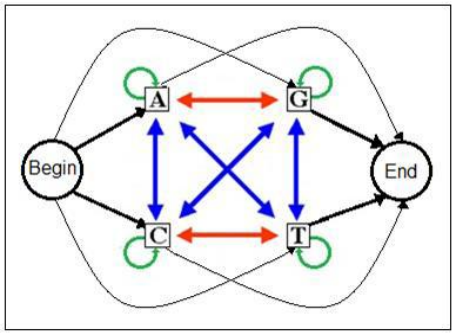


Fig. 10.1.2. Introducerea stărilor "Begin" şi "End" la lanțul Markov pentru analiza secvențelor ADN

D. Exemplu

Să calculăm probabilitatea de a găsi secvența „cggt" la începutul unui lanț ADN.

Înlocuind în formula (10.1.2.d') obținem:

$$P(cggt) = P(c|B) \times P(g|c) \times P(g|g) \times P(t|g) \times P(E|t) \quad (10.1.2.h)$$

Aceste probabilități sunt cunoscute; pentru porțiuni care nu sunt „insule CpG" (pe care le vom discuta în acest capitol) aceste valori sunt:

$P(c|B) = 0.13,\ P(g|c) = 0.08,\ P(g|g) = 0.30,\ P(t|g) = 0.21$

De obicei sfârşitul secvenței nu este introdus în model presupunându-se că secvența se poate termina oriunde; sfârşitul se introduce doar dacă trebuie considerată explicit lungimea frecvenței.

10.2. Estimarea parametrilor modelului

10.2.1. Punerea problemei

În cazul modelării secvențelor prin lanțuri Markov, este evident că probabilitățile tranzițiilor ocupă o poziție centrală.

A. Enunțul problemei: Fiind dat un set de date, D (de exemplu, o succesiune de secvențe), să se determine probabilitățile tranzițiilor.

Formal problema este determinarea parametrilor ϑ care maximizează $P(D|\vartheta)$, adică face setul de date D cel mai probabil.

Există mai multe abordări posibile ale problemei: fie prin metoda lui Bayes a probabilităților condiționate, fie prin metoda asemănării maxime. În cele ce urmează vom prezenta metoda asemănării maxime (maximum likelyhood).

După cum se remarcă uşor din formulele prezentate mai sus, vom avea de folosit în mod frecvent produsul unor probabilități, iar aceste probabilități se vor obține din analiza unor secvențe care constituie setul de date D. Ce ne facem însă în cazul în care vreun simbol nu apare? În acest caz probabilitatea sa va fi zero şi întregul produs se anulează! Desigur, astfel de situații ar fi rare, însă nu imposibile, mai ales când setul de da6te este mai restrâns. Pentru astfel de situații vom apela la proceduri prin care să aproximăm valorile probabilităților, evitând însă valoarea zero, aproximații numite „estimate Laplace".

Să pornim însă cu abordarea clasică a exprimării probabilității de apariție a unui simbol.

B. Probabilități individuale

(i) Pornim de la cazul simplu în care setul de date este compus dintr-un anumit număr de secvențe.

Pentru moment să facem următoarele asumpții:
- fiecare poziție este independență de celelalte
- fiecare poziție este generată de aceeaşi distribuție multinominală.

(ii) În cazul în care ne referim la acizi nucleici, vom avea 4 componente (nucleotide) ale căror probabilități dorim să le exprimăm. În cazul ADN, vom reprezenta nucleotidele prin simbolurile „a, c, g, t", deci dorim să exprimăm parametrii *P(a), P(c), P(g), P(t)*.

(iii) Să considerăm un set de secvențe şi să notăm cu n_a numărul de apariții a simbolului „a" în aceste secvențe. Atunci:

$$P(a) = n_a / \sum n_i \qquad (10.2.1.a)$$

unde indicele i ia valorile „a, c, g, t", adică numitorul reprezintă de fapt numărul total al elementelor din toate secvențele.

$$\sum n_i = n_a + n_c + n_g + n_t \qquad (10.2.1.b)$$

În mod similar vom avea relații pentru probabilitățile celorlalte tipuri de nucleotide: „c, g şi t".

(iv) Să luăm un exemplu.

Fie secvențele:
X: accgcgctta
Y: gcttagtgac
Z: tagccgttac.

Am luat aici secvențe egale ca lungime, însă la modul general ele au lungimi diferite. Observăm că putem calcula uşor probabilitățile celor 4 tipuri de nucleotide:

P(a) = 6/30 = 0.2 P(g) = 7/30 = 0.233

P(c) = 9/30 = 0.3 P(t) = 8/30 = 0.267

unde numitorul satisface relația (10.2.1.b), adică 30 = 6+9+7+8.

10.2.2. Estimarea Laplace

A. Să continuăm exemplul anterior, însă într-o versiune în care un simbol dispare, de exemplu, să considerăm că în aceste secvențe toate nucleotidele „a" au fost substituite cu „g". Secvențele ar arăta acum astfel:

X': gccgcgcttg
Y': gcttggtggc (10.2.2.a)
Z': tggccgttgc.

Acum probabilitățile au devenit:

P(a) = 0/30 = 0 P(g) = 13/30 = 0.433

P(c) = 9/30 = 0.3 P(t) = 8/30 = 0.267

B. Estimarea Laplace se face provenind de la cunoştinţele anterioare ("beliefs") prin care presupunem că totuşi fiecare component are o probabilitate nenulă de apariție. În acest scop von introduce la numărătoare o unitate, care în realitate nu apare în secvențe, numită "pseudocount". Dar, pentru a nu favoriza numai un component, vom acorda acest "pseudocount" pentru toate componentele.

C. Formal acest lucru se exprimă astfel:

$$P(a) = (n_1 + 1)/\sum(n_i + 1) \qquad (10.2.2.b)$$

D. Reluând calculele de mai sus vom obține:

P(a) = (0+1)/(30+4) = 1/34 = 0.029

P(c) = (9+1)/34 = 10/34 = 0.294

P(g) = 14/34 = 0.412

P(t) = 9/34 = 0.264

Vedem că valorile de aici sunt o bună aproximație a celor reale, evitând valoare 0.

10.2.3. Estimarea Laplace generalizată

A. În varianta descrisă mai sus am introdus doar câte „1" pseudocount pentru fiecare simbol. Putem generaliza formula (10.2.m) introducând un număr „m" de pseudocounts, adică instanțe virtuale:

$$P(a) = (n_a + p_a m)/[(\sum n_i) + m] \qquad (10.2.3.a)$$

unde p_a reprezintă „probabilitatea anterioară" ("prior probability") a lui „a".

B. De exemplu, putem pentru P(c), în cazul m = 8 şi $p_c = 0.25$

P(c) = (9+0.5×8)/(30+8) = 11/38 = 0.289.

10.2.4. Estimarea parametrilor de ordin I

A. După ce am prezentat estimarea Laplace a probabilităților individuale, putem reveni la punctul de vedere prezentat la începutul descrierii lanțurilor Markov, adică să ținem cont de faptul că probabilitatea apariției într-o secvență a unui simbol depinde de pozițiile anterioare. În cazul în care vom lua în considerare doar 1 poziție anterioară, probabilitățile calculate le vom numi probabilități de ordin 1. Cu această convenție, probabilitățile individuale independente s-ar numi probabilități de ordin 0.

B. Probabilități Laplace de ordin 1.

Definiție: Probabilitatea de ordin 1 este probabilitatea unui element, „t", de a fi precedat de un anumit element, „s", adică probabilitatea secvenței „st".

$$P(t|s) = (n_{st} + 1)/(\sum n_{si} + 1) \qquad (10.2.4.a)$$

unde i ia toate valorile posibile ale simbolurilor.

C. Exemplu

Fie din nou secvențele din (10.2.2.a)

X': gccgcgcttg
Y': gcttggtggc
Z': tggccgttgc.

Iată un model de calcul aplicând (10.2.4.a):

$P(c|g) = (7+1)/(12+4),\ n_{gc} = 7$

$$n_{ga} + n_{gc} + n_{gg} + n_{gt} = 0 + 7 + 3 + 2 = 12$$

Valorile care se obțin pentru exemplul nostru sunt:

P(a|g) = (0+1)/(12+4)	*P(a|c) = (0+1)/(7+4)*
P(c|g) = (7+1)/(12+4)	*P(c|c) = (2+1)/(7+4)*
P(g|g) = (3+1)/(12+4)	*P(g|c) = (3+1)/(7+4)*
P(t|g) = (2+1)/(12+4)	*P(t|c) = (2+1)/(7+4)*
P(a|a) = (0+1)/(0+4)	*P(a|t) = (0+1)/(8+4)*
P(c|a) = (0+1)/(0+4)	*P(c|t) = (0+1)/(8+4)*
P(g|a) = (0+1)/(0+4)	*P(g|t) = (5+1)/(8+4)*
P(t|a) = (0+1)/(0+4)	*P(t|t) = (3+1)/(8+4)*

10.2.5. Lanțuri Markov de ordin superior

A. Generalizând definiția lanțului Markov de ordin 1 se pot defini lanțuri Markov de ordin superior. Deoarece un lanț Markov este rezultatul unui proces prin care se realizează secvența analizată, se mai foloseşte şi termenul de ,,proces Markov".

B. Definiție: Un proces Markov de ordin n este un proces stochastic în care fiecare eveniment depinde de n evenimente precedente, adică un simbol depinde de n simboluri anterioare.

Formal putem scrie acest lucru prin relația:

$$P(x_i|x_{i-1}, x_{i-2}, \dots x_1) = P\left(x_i|x_{i-1}, \dots x_{i-n}\right) \qquad (10.2.5.a)$$

adică din întreaga secvență din față $(x_1x_2\dots x_{i-1})$ s-a reținut doar porțiunea de n elemente care determină următorul (al n+1 -lea).

C. Numărul de parametrii ce trebuie determinați creşte exponențial cu ordinul lanțului. Am văzut că pentru ADN, unde alfabetul are doar 4 simboluri, numărul de parametrii (probabilități) pentru ordinul 0 era 4, iar pentru ordinul 1 era 4^2. În general, pentru ordinul ,,n" numărul de probabilități va fi 4^{n+1}.

D. Aplicație – lanțuri Markov de ordinul 5

O problemă importantă în utilizarea lanțurilor Markov este alegerea ordinului. Practic s-a găsit că ordinul 5 este cel mai potrivit pentru găsirea genelor, procedură folosită de către programul **GeneMark**, realizat de către Borodovski.

Să remarcăm în încheiere că se pot efectua analize secvențiale şi cu lanțuri Markov neomogene, în care se consideră diferite distribuții în diferite regiuni de secvență.

10.3. "Open Reading Frames" (ORF)

10.3.1. Noţiunea de „cadru de citire"

A. Una dintre aplicaţiile majore ale proceselor Markov este identificarea regiunilor care codifică sinteza unei protein, adică găsirea „genelor".

Definiţie: o secvenţă ADN care ar putea codifica o proteină se numeşte "Open Reading Frame" – ORF (cadru de citire).

B. Un ORF are următoarele proprietăţi:
- începe cu un codon de start (ATG),
- se termină cu un codon de stop (TAG, TAA, TGA),
- nu are codon de stop intern,
- satisface anumite cerinţe minime de lungime.

Fig. 10.3.1.a. Secvenţe potenţial ORF

C. Este cunoscut faptul că un codon este format din 3 nucleotide, deci unei secvenţe de 3 nucleotide îi corespunde un aminoacid.

O problemă esenţială în înţelegerea procesului de sinteză a proteinelor este modul în care este iniţializat procesul. Dintr-o secvenţă de ADN, formată din două lanţuri complementare, putem – în principiu – să efectuăm 6 variante de citire (figura 10.3.1.b)

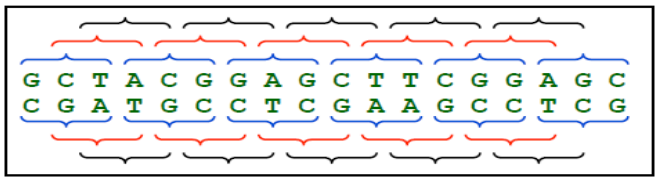


Fig. 10.3.1.b. Codificări posibile ale unei proteine dintr-un lanţ de ADN

D. Exemplu

Să găsim un ORF în secvenţa de mai jos şi să stabilim structura polipeptidului ce poate fi sintetizat.

Fie secvenţa:

CTGGATGATTCAGCTGGTCTAGTGACTAGC

Vom proceda după cum urmează:
- aplicăm o fereastră de căutare cu lăţimea unui codon, adică 3 nucleotide, pe care o deplasăm poziţie cu poziţie până când găsim un codon de start (care corespunde metioninei)

CTGG**ATG**ATTCAGCTGGTCTAGTGACTAGC

- împărţim restul lanţului în secvenţe de câte 3 elemente (în codoni)

– urmărim codonii de-a lungul lanțului până când întâlnim un codon de stop

CTGG **ATG** ATT CAG CTG GTC **TAG** TGACTAGC

– în final asociem fiecărui codon aminoacidul corespunzător conform codului genetic, adică avem pentapeptidul:

Met-Ile-Gln-Leu-Ala

sau, utilizând simbolurile din Bioinformatică:

M I Q L A.

10.3.2. Metode pentru găsirea genelor

A. Un obiectiv important în bioinformatică este găsirea genelor ce determină anumite caracteristici inclusiv gene implicate în procese patologice.

Prin metodele prezente până acum putem vedea că avem o paletă mai largă, care cuprinde:

a) căutarea prin similaritate de secvență (sss – Search by Sequence Similarity), adică se caută prin analiză secvențială potriviri cu secvențe cunoscute a fi legate de gene

b) căutare prin semnal (Search by Signal), metodă bazată pe identificarea „semnalelor" implicate în expresia genei (v. semnalizare celulară)

c) căutare prin conținut (Search by Content) – găsirea genelor prin proprietățile statistice ce disting ADN-ul ce codifică proteinele (exoni) de cel ce nu codifică (introni).

B. Căutarea prin metode markoviene, cu identificare de ORF-uri este inclusă în metoda căutării prin conținut. Să mai adăugăm următoarele observații:

– anumiți AA apar mai frecvent în exoni decât în introni (exemplu: Leu mai frecvent ca Trp)
– diferiți AA au un număr diferit de codoni (de exemplu, pentru Trp avem doar 1 codon, în timp ce pentru Leu sunt 6 codoni)
– pentru AA care sunt codificați prin mai mulți codoni, unii codoni apar mai frecvent decât alții, această „preferință" pentru codoni variază cu specia.

C. În tabelul 10.3.2 sunt trecute comparative câteva frecvențe de codificare ale unor AA.

Tabelul 10.3.2. Frecvențe de codificare a unor aminoacizi prin diferiți codoni

AA	codon	freq per 1000			
Gly	GGG	1.89	Glu	GAG	15.68
Gly	GGA	0.44	Glu	GAA	57.20
Gly	GGU	52.99	Asp	GAU	21.63
Gly	GGC	34.55	Asp	GAC	43.26

11. Modele Markov Ascunse

11.1. Suportul biologic al abordării

11.1.1. Insulele CpG

A. Un aspect interesant găsit în studiile privind frecvența unor microsecvențe de câte două nucleotide se referă la așa-numitele „insulele CpG”.

B. În cursul proceselor biologice se poate întâmpla ca citozina (C) să fie metilată și substituția sa cu timină (T) devine mult mai probabilă, acest fenomen fiind mai frecvent când citozina este urmată secvențial de guanină. Această succesiune citozină-guanină într-un lanț ADN se notează „CpG”, pentru a face distincție între această succesiune și notația CG folosită pentru a desemna perechea citozină-guanină din cele două lanțuri ale dublului helix ADN, menținut prin punțile de hidrogen.

C. Iată însă că natura a găsit o cale de a se apăra împotriva acestei probabilități ridicate de mutație, prin scăderea frecvenței microsecvențelor CpG în lanțurile ADN. În tabelul 11.1.1, secțiunea din dreapta, marcată cu semnul „-” observăm că frecvența reală a perechii „CG” este de 0.078, de cca. 4 ori mai mică decât o frecvență aleatoare (0.25)!

Tabelul 11.1.1. Secțiunea din stânga, notată cu „+” reprezintă frecvența perechilor de 2 nucleotide în „insulele CpG”, iar secțiunea din dreapta, notată „-” conține secvențele în regiunile neinsulare. Pe axa verticală s-a notat primul nucleotid din pereche, iar pe cea orizontală, al doilea.

P(*c* | *a*)

+	*a*	*c*	*g*	*t*
a	.18	.27	.43	.12
c	.17	.37	.27	.19
g	.16	.34	.38	.12
t	.08	.36	.38	.18

CpG

-	*a*	*c*	*g*	*t*
a	.30	.21	.28	.21
c	.32	.30	.08	.30
g	.25	.24	.30	.21
t	.18	.24	.29	.29

null

D. Un alt aspect interesant este că în alte regiuni ale lanțurilor ADN, această „protecție” nu mai este menținută - este vorba mai ales de regiunile care au alte mecanisme de protecție, cum ar fi zonele „promoter”, care conțin secvențele de inițializare a proceselor de sinteză, zone care sunt de obicei controlate prin cuplarea unor proteine cu rol de control al mecanismelor de sinteză. Aceste regiuni în care frecvența perechilor microsecvențiale CG este apropiată de cea naturală, poartă denumirea de „insulele CpG”. Studiul a 48 astfel de insule a permis stabilizarea probabilității secvențelor din tabelul 11.1.1.

11.1.2. Distincția „stare” - „simbol”

A. Remarcăm deci, un fapt esențial: în secvențele reale ADN întâlnim regiuni cu probabilități diferite ale microsecvențelor – perechi de nucleotide succesive. Vom numi aceste regiuni - „stări”, iar apariția/generarea unui simbol va fi numită „emisie”. Este

evident că probabilitatea de „emisie” a unui simbol va fi dependentă de „starea” în care se găseşte sistemul.

Din punct de vedere formal, emisia unui simbol este echivalentă cu „tranziția” definită în descrierea unui lanț Markov.

B. Vom introduce deci două feluri de probabilități:

a) probabilitatea ca sistemul să treacă dintr-o stare în alta; să notăm cele două stări *„l”* şi *„k”,* starea inițială fiind k şi cea după „tranziție” fiind l. Probabilitatea de tranziție $k \rightarrow l$ va fi dată de relația:

$$a_{kl} = P(\pi_i = l | \pi_{i-1} = k) \qquad \text{(II.1.2.a)}$$

unde π_i şi π_{i-1} reprezintă probabilitatea stării i, respectiv a stării anterioare i-1; să mai menționăm că probabilitatea de a rămâne în aceeaşi stare este $1\text{-}a_{kl}$.

b) probabilitatea de „emisie” a unui simbol „b” când sistemul este în starea „k”:

$$e_k(b) = P(x_i = b | \pi_i = k) \qquad \text{(II.1.2.b)}$$

11.2. Enunțul problemei în HMM

11.2.1. Denumirea HMM (Hidden Markov Model)

A. În modelele Markov clasice există o singură stare, cunoscută, şi problema era să se determine probabilitatea emisiei unui simbol în funcție de simbolul emis anterior.

B. În cazul nostru problema are un aspect nou: simbolul *(„b”)* poate fi emis fie în starea *„k”,* fie în starea *„l”,* cu probabilități diferite. Problema este trivială în cazul în care cunoaştem starea. Devine însă foarte interesantă problema de a determina starea în care a fost sistemul când a emis simbolul. Datorită faptului că tocmai starea este necunoscută când se generează o succesiune de simboluri fiind posibilă şi trecerea sistemului – tranziția – dintr-o stare în alta, a generat denumirea de „model Markov ascuns” (Hidden Markov Model), în sensul că starea în care s-a emis un element a fost necunoscută (ascunsă).

11.2.2. Reprezentarea schematică a unui HMM

A. Să considerăm un exemplu în care sistemul poate intra fie în starea „1”, fie în „2”, fiecare cu probabilități specifice de emisie a celor 4 simboluri dintr-un lanț ADN.

După emiterea unui simbol, sistemul poate fie să rămână în aceeaşi stare, (1, respectiv 2), cu probabilități cunoscute, fie trecerea în noi stări: 3 din 1, sau 4 dacă starea anterioară a fost 2.

Din nou, sistemul emite către un simbol, după care poate rămâne în aceeaşi stare (3 sau 4), sau poate încheia emisia, trecând în starea 5 (end) – figura 11.2.2.

B. Problema principală este: determinarea succesiunii stărilor, pornind de la o secvență emisă.

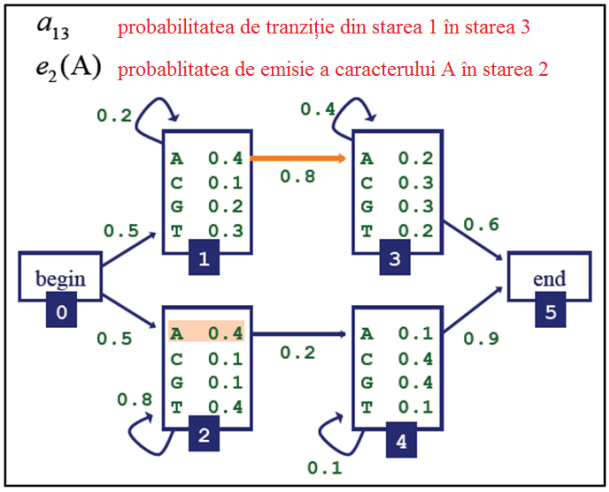


Fig. 11.2.2. Exemplu de HMM

11.3. Algoritmi de calcul pentru HMM

Există mai multe metode prin care putem aborda problema detecției succesiunii stărilor care au generat o frecvență dată, cunoscute fiind atât probabilitățile de tranziție între stări, precum şi probabilitățile de emisie ale tuturor simbolurilor, pentru toate stările.

a) Algoritmul "Forward" – prin care se calculează cât de probabilă este o secvență dată;

b) Algoritmul "Viterbi" prin care se determină care este cea mai probabilă „cale" (secvență de stări ascunse = path) pentru a genera secvența dată;

c) Algoritmul Baum – Welch ("Forward - Backward"), care încearcă să răspundă la întrebarea: „cum putem afla parametrii modelului HMM dintr-un set de secvențe date?"

În cele ce urmează vom prezenta în detaliu doar algoritmul Viterbi – cel mai folosit.

11.3.1. Algoritmul Viterbi

A. Algoritmul propus de Andrew Viterbi (1967) porneşte de la nişte presupuneri simplificatoare:

- evenimentele observate precum şi cele ascunse sunt într-o secvență, ce corespunde – cel mai adesea – cu o succesiune temporală,
- cele două secvențe – a evenimentelor observate, respectiv a stărilor – trebuie să fie aliniate şi unei instanțe de eveniment observat îi corespunde doar unei instanțe de stare (eveniment ascuns),
- calculul celei mai probabile căi a secvenței stărilor până la un moment „t" depinde numai de evenimentul observat la momentul t şi de cea mai probabilă secvență până la momentul anterior t-1.

Aceste condiții sunt în general îndeplinite în modelele HMM de ordinul 1.

B. Din punct de vedere formal, păstrăm notațiile din formulele (II.1.2.a şi b) şi mai notând:

- $x_0,\ldots,x_t$ = şirul de observații
- $y_0,\ldots,y_t$ = secvența stărilor - cea mai probabilă să fi produs observațiile.

Cu aceste notații putem scrie o relație de recurența pentru probabilitatea $V_{t,k}$ a celei mai probabile secvența de stări responsabile pentru primele $t + 1$ observații (am început indexarea de la 0), având k drept stare finală:

$$V_{t,k} = P(x_t|k) \times \max_{y \in Y}\left(a_{y,k} \times V_{t-1,y}\right), \text{ cu}$$

$$V_{0,k} = P(x_0|k) \times \pi_k \qquad \text{(II.3.1.a)}$$

Observăm că produsele succesive de probabilități ; totdeauna subunitare, vor genera numere din ce in ce mai mici, astfel încât s-a propus exprimarea într-un spațiu logaritmic, calculându-se $log\,[V_{t,k}]$.

11.3.2. Exemplu

A. Considerăm două persoane, A şi B (Ann si Bob) care stau în localități diferite dar comunică zilnic. Bob îi transmite activitățile dominante din fiecare zi, care pot fi: (plimbare/walk, cumpărături/shopping sau curățenie/clean) prescurtat (w, s, c). Probabilitatea fiecărei activități depinde de vremea de afară, adică „starea" sistemului, care poate fi {ploioasă/rainy sau însorită/sunny (R, S)}. Din comunicările telefonice, Ann ştie că Bob a avut în trei zile succesive acțiunile: (w, s, c).

B. Problema ar fi: să se determine „starea" cea mai probabilă în fiecare din cele trei zile. Se consideră cunoscute probabilitățile stărilor R („Ploaie"), respective S („Soare") în regiunea respectivă. Totodată se mai consideră cunoscute şi probabilitățile de inițiere a oricărei acțiuni în diferite condiții meteorologice – figura 11.3.2.a.

C. Problema: determinarea succesiunii stărilor cunoscând o secvență output („emisii"). Notații: Stări (R,S), Emisii (w, s, c).

Rezolvare:

Ziua 1: $p'_1(w) = p(R) \times p(w|R) = 0.6 \times 0.1 = 0.06$
$p''_1(w) = p(S) \times p(w|S) = 0.4 \times 0.6 = 0.24$

Ziua 2: $p'_2(s|w) = p'_1(w) \times p(R|R) \times p(s|R) = 0.06 \times 0.7 \times 0.4 = 0.0168$
$p''_2(s|w) = p'_1(w) \times p(S|R) \times p(s|S) = 0.06 \times 0.3 \times 0.3 = 0.0054$
$p'''_2(s|w) = p''_1(w) \times p(R|S) \times p(s|R) = 0.24 \times 0.4 \times 0.4 = 0.0384$
$p''''_2(s|w) = p''_1(w) \times p(S|S) \times p(s|S) = 0.24 \times 0.6 \times 0.3 = 0.0432$

Ziua 3: $p(c|ws) = p'''_2(s|w) \times p(R|R) \times p(c|R) = 0.0384 \times 0.7 \times 0.5 = 0.01344$
$p(c|ws) = p'''_2(s|w) \times p(S|R) \times p(c|S) = 0.0384 \times 0.3 \times 0.1 = 0.001152$
$p(c|ws) = p''''_2(s|w) \times p(R|S) \times p(c|R) = 0.0432 \times 0.4 \times 0.5 = 0.00864$
$p(c|ws) = p''''_2(s|w) \times p(S|S) \times p(c|S) = 0.0432 \times 0.6 \times 0.1 = 0.002592$

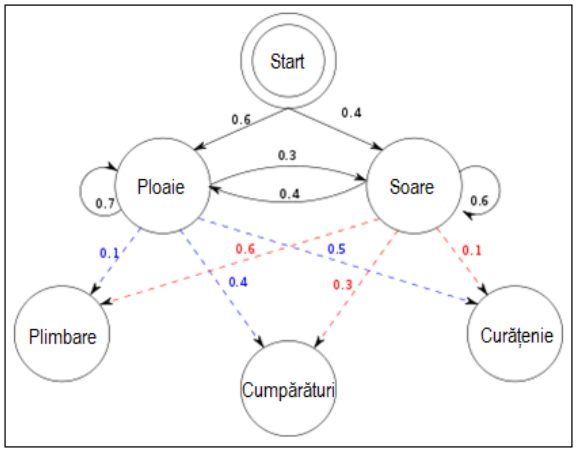


Fig. 11.3.2.a. Schema HMM pentru exemplul 11.3.2.

D. Vom alege acum valoarea cea mai mare pentru probabilitatea ultimei stări (este 0.1344 – corespunzătoare acțiunii de curățenie într-o zi ploioasă, efectuată după o zi de cumpărături într-o zi ploioasă, după ce anterior a fost la plimbare într-o zi însorită). Deci parcurgem drumul înapoi, exact ca la orice "trace back" din algoritmii de programare dinamică.

Acest lucru poate fi ilustrat prin nişte diagrame specifice, numite "trellis": o astfel de diagramă este prezentată în figura 11.3.2.b.

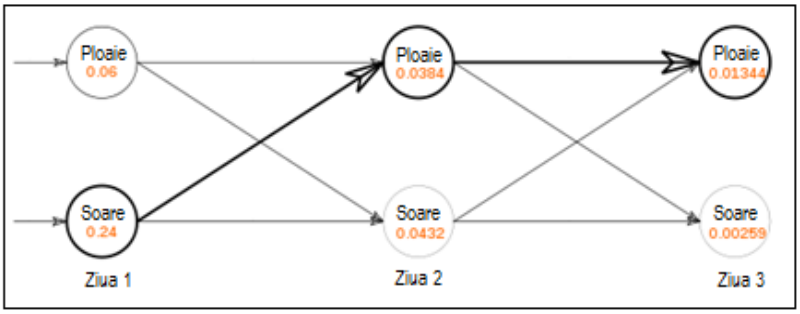


Fig. 11.3.2.b. Diagrama "trellis" pentru exemplul 11.3.2

11.4. Aplicarea HMM pentru discriminare

11.4.1. Notații

A. Modelele Markov Ascunse, HMM, permit identificarea succesiunii stărilor posibile care au generat o anumită secvență. Această capacitate poate fi utilizată pentru recunoaşterea/discriminarea diferitelor regiuni din secvențele analizat.

Am menționat anterior (§ 11.1.1) existența „insulelor CpG", iar în tabelul 11.1.1 am prezentat şi probabilitățile tuturor perechilor de nucleotide în cele două tipuri de secvențe de ADN – din insule CpG (modelul notat cu „+"), respectiv din afara insulelor (modelul numit "null" şi notat cu „-").

B. Notațiile folosite uzual în aceste analize sunt:

– probabilitatea de „tranziție în interiorul fiecărui model", notată a^{+}_{st} pentru insulele CpG (modelul „+"), respectiv a^{-}_{st} înafara insulelor:

$$a^{t}_{st} = c^{t}_{st} / \sum_{t'} c^{t}_{st'} \qquad \text{(II.4.I.a)}$$

unde c^{+}_{st} arată de câte ori simbolul t a fost urmat de simbolul s (t şi s au valori din alfabetul de 4 litere: A, C, G, T).

O relație similară se scrie pentru a^{-}_{st}.

De fapt, valorile din tabelul 11.1.1 sunt chiar acestea.

C. Observăm asimetria tablourilor: A urmat de G este mult mai frecvent decât G urmat de A, mai ales în insule, iar T urmat de A apare mult mai rar decât GA şi în insule şi în afara lor.

11.4.2. Scoruri "log-odd"

A. Pentru a utiliza modelele Markov pentru discriminarea diferitelor regiuni se introduce un scor, numit "raport log-odd", S(x), dat de relația:

$$S(x) = \log \frac{P(x|model"+")}{P(x|model"-")} = \sum_{i=1}^{L} \log \frac{a^{+}_{x_{i-1}x_i}}{a^{-}_{x_{i-1}x_i}} = \sum_{i=1}^{L} \beta_{x_{i-1}\,x_i} \qquad \text{(II.4.I.b)}$$

în care x este o secvență de două simboluri, iar β este logaritmul raportului probabilităților corespunzătoare tranzițiilor în cele două modele, „+" şi „-".

Folosind logaritmi în baza 2, raportul β se va măsura în biți.

B. Pentru discriminarea insulelor CpG față de celelalte regiuni, valorile lui β sunt cele din tabelul 11.4.2. prezentat mai jos:

Tabelul 11.4.2. Valorile lui β pentru discriminarea CpG

β	A	C	G	T
A	-0.740	0.419	0.580	-0.803
C	-0.913	0.302	1.812	-0.685
G	-0.624	0.461	0.331	-0.730
T	-1.169	0.573	0.393	-0.679

C. Făcând o reprezentare grafică a distribuției scorurilor (nominalizate în funcție de lungimea secvențelor), se observă o discriminare rezonabilă între regiunile CpG față de celelalte regiuni (figura 11.4.2)

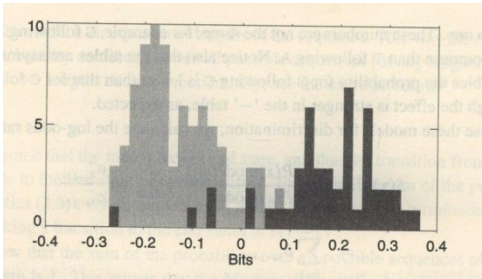


Fig. 11.4.2. Histograma scorurilor normalizate după lungime. Insulele CpG sunt reprezentate cu negru, iar celelalte regiuni cu gri

11.5. Modele Markov ascunse cu inserții şi deleții

11.5.1. Stări silențioase

Într-un model HMM am întâlnit până acum două stări în care sistemul nu emitea nici un simbol erau Begin şi End. Modelul poate fi generalizat prin introducerea unor stări de acest fel oriunde în lanț.

11.5.2. Schema HMM generală

Să notăm cu M_j stările succesive prin care trece sistemul, reprezentate prin pătrate, apoi I_j inserțiile, reprezentate în romburi şi cu D_j delețiile, plasate în cercuri, vom obține o schemă generală de model Markov ascuns, aplicabil şi în analiza secvențială multiplă modelul propus de Haussler, 1993 (figura 11.5.2).

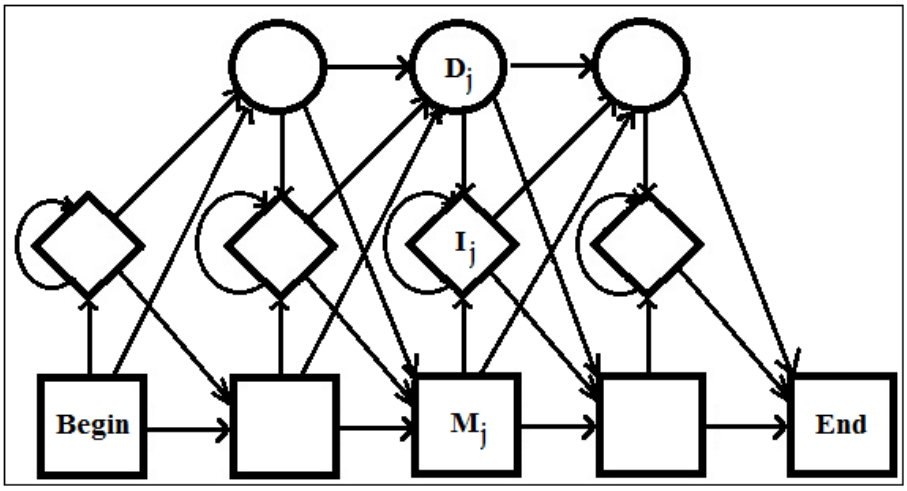


Fig. 11.5.2. Structura generală a unui profil HMM

12. Analiza filogenetică

12.1. Inferență filogenetică

12.1.1. Obiectivul analizei filogenetice

A. Unul dintre cele mai importante capitole ale bioinformaticii este reprezentat de analiza filogenetică. Am văzut şi în capitolele anterioare că în cursul evoluției au loc diferite evenimente - deleții şi substituții, însoțite de diverse modificări ale proprietăților moleculelor implicate. Uneori aceste modificări sunt minore, se conservă proprietățile importante precum şi funcția în mecanismele celulare. Alteori aceste modificări sunt mai mari, pot fi reflectate şi la nivel fenotipic şi/sau funcțional, fiind considerate mutații. Analiza originii, evoluției şi relațiilor între diferitele molecule reprezintă obiectul analizei filogenetice.

B. Să mai menționăm că relațiile evolutive sunt importante nu numai pentru căutarea ancestorilor comuni, ci permit o mai bună abordare a alinierii multiple. În acest sens am şi pomenit de metoda arborilor pentru ierarhizarea moleculelor la introducerea în alinierea progresivă.

C. În cele ce urmează vom aborda analiza filogenetică prin construcția arborilor filogenetici, pornind de la câteva noțiuni introductive despre arbori şi apoi ne vom concentra asupra principalelor două metode de construcție a arborilor: metoda distanțelor şi metoda parsimoniei, trecând sumar şi prin metoda asemănării maxime.

12.2. Noțiuni generale despre arbori

12.2.1. Terminologie

A. Biologia moleculară a adus argumente incontestabile privind similaritatea mecanismelor moleculare din materia vie, care sugerează că toate organismele vii de pe pământ, din toată istoria lumii, trebuie să fi avut un ancestor comun. Deci între oricare două organisme/specii se pot găsi diverse grade de rudenie. Relațiile de înrudire între specii se numeşte filogenie. Aceste relații pot fi reprezentate printr-un arbore filogenetic.

B. Din punct de vedere istoric, prima lucrare care sugerează extragerea informației din secvențe moleculare pentru a studia filogenia speciilor a fost publicată de Zuckenkandl şi Pauling [1962]. Apoi Langley şi Fitch [1974] arată că proteinele se modifică cu rate diferite, iar Doolittle [1996] demonstrează o bună corelație între lungimi şi perioadele de timp de evoluție.

12.2.2. Utilitate, motivați

A. Studiul arborilor filogenetici are o mare însemnătate teoretică şi practică:

- putem înțelege mai bine relațiile evolutive ale speciilor,
- putem înțelege cum au evoluat diverse funcții,
- arborii filogenetici aduc informații pentru alinierea multiplă,

- se pot identifica cele mai importante elemente, porțiunile care se conservă în unele clase de secvențe.

B. Exemple

În figura 12.2.2.a este prezentat un arbore filogenetic *al genelor* ce codifică informații privind globinele.

În figura 12.2.2.b este o schemă simplificată a arborelui *speciilor* de babuini.

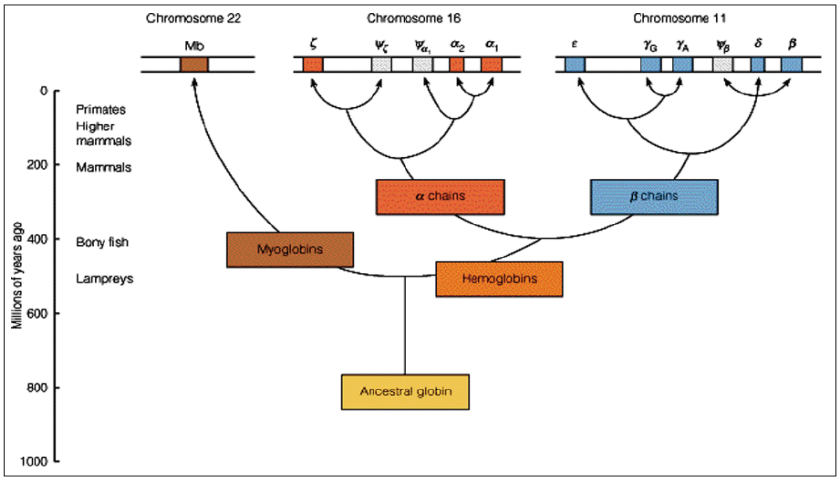


Fig. 12.2.2.a. Arbore filogenetic pentru gene: globinele

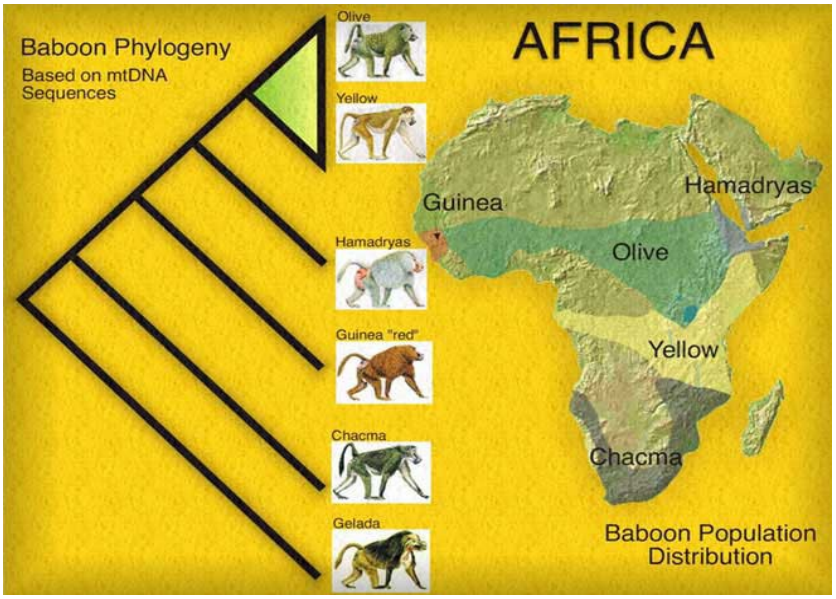


Fig. 12.2.2.b. Arbore filogenetic pentru specii: babuinii

12.3. Proprietățile arborilor

12.3.1. Definiție, structură

A. Definiție: un arbore (“tree”) este un graf aciclic, nedirecționat.

B. Elementele componente ale uni arbore sunt:

- frunze (“leaves”) - elemente primare, obiecte (de exemplu: specii, gene, secvențe de proteine, chiar componente ale unei secvențe) cu poziție terminală în arbore; se mai numesc şi noduri exterioare sau noduri de grad „1”; de obicei sunt notate cu numere (uneori cu litere)
- noduri (“nodes”) - reprezintă intersecții de ramuri; se numerotează de la n+1 în sus (în cazul unui arbore cu n frunze)
- ramuri (“edges”) - sunt legăturile dintre noduri; deseori au asociată o „lungime” calculată după diverse criterii.

Observație: în cazul arborilor în care se prezintă relații între specii, o frunză se mai numeşte “taxon” (pl: taxa)!

12.3.2. Tipuri de arbori

A. Clasificare Arborii filogenetici pot fi de două tipuri:

a) arbori fără rădăcină (“unrooted tree”) - în care se specifică doar relațiile dintre noduri (figura 12.3.2.a)

b) arbori cu rădăcină (“rooted tree”) - rădăcina este ultima ramură ce ajunge la ultimul nod, presupus a reprezenta ancestorul comun (figura 12.3.2.b).

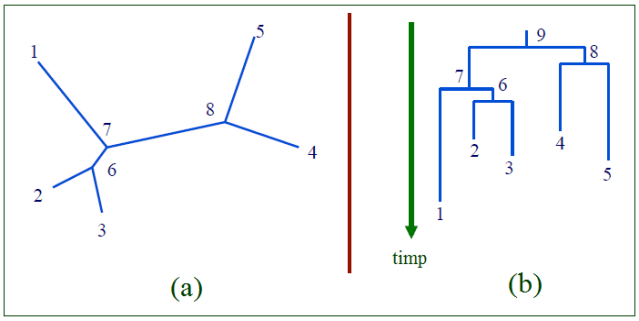


Fig. 12.3.2. Tipuri de arbori

B. Pentru arborii cu rădăcină se consideră convențional un sens al evoluției în timp, uzual plasând frunzele în partea de jos iar ancestorul comun (rădăcina) în partea superioară. De asemenea, lungimile ramurilor ar trebui să fie proporționale cu „distanța” dintre noduri (similară distanțelor dintre secvențe). Însă în arborii fără rădăcină nu avem un sens de desfăşurare (nu avem o axă a timpului), şi nici lungimile nu trebuie să fie dependente de gradul de similaritate (deşi uneori se practică o reprezentare aproximativ proporțională). De aceea spunem despre arborii fără rădăcină că reprezintă topologia arborelui.

C. Inferență filogenetică

Operațiunea principală (concluzia) studiilor filogenetice, ce reprezintă stabilirea unui arbore filogenetic care caracterizează linia evolutivă între specii sau gene se numeşte „inferență filogenetică”.

12.3.3. Numărul de arbori

A. Numărul de noduri şi ramuri

Pentru arborii cu rădăcină

Considerăm un arbore cu n frunze de la care pleacă în sus n ramuri. Unirea a fiecare două frunze va genera un nod de ordin superior şi va reduce numărul ramurilor cu o unitate. Vom avea deci 2n-1 noduri şi 2n-2 ramuri (dacă nu mai numărăm ultima ramură - cea de deasupra nodului rădăcină).

Pentru arborii fără rădăcină cu n frunze vom avea 2n-2 noduri şi 2n-3 ramuri.

B. Calculul numărului de arbori fără rădăcină N_{AFR}

Pornind de la cel mai simplu arbore fără rădăcină – cel cu 3 frunze (figura12.3.3.a), vedem că putem genera din el 3 variante de arbori cu 4 frunze asociind a 4-a frunză pe oricare dintre cele 2×3-3=3 ramuri (figura 12.3.3.b). Aceşti arbori cu 4 frunze au 2×4–3=5 ramuri.

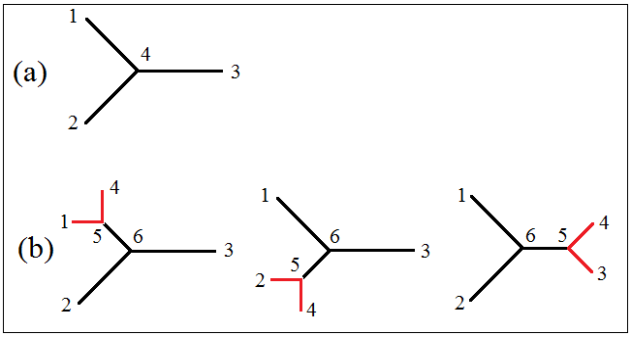


Fig. 12.3.3.a si b. Formarea incrementală a arborilor fără rădăcină

Deci pentru n = 4 frunze, pe cei 3 arbori fiecare cu 2n-3 ramuri putem adăuga a 5-a frunză în 3×5=15 poziții diferite. Putem stabili o relație de recurență cu datele din tabelul 12.3.3.a.

Tabel 12.3.3.a. Numărul arborilor fără rădăcină

Nr. frunze	3	4	5	…	n
Nr. noduri	4	6	8	…	2n – 2
Nr. ramuri	3	5	7	…	2n – 3
Nr. arbori	1	3×5	3×5×7	…	(2n – 5)!!

Putem sintetiza aceste date în relația (12.3.3.a):

$$N_{AFR} = \prod_{i=3}^{n}(2i-5) = (2n-5)!! \qquad (12.3.3.a)$$

C. Calculul numărului de arbori cu rădăcină N_{AR}

Putem face un raționament similar cu cel de mai sus, pornind tot de la cel mai simplu arbore fără rădăcină, cel cu 3 frunze (figura 12.3.3.c). Vom putea din nou obține 3 variante de arbori cu rădăcină, plasând-o pe fiecare dintre cele 3 ramuri (figura 12.3.3.d).

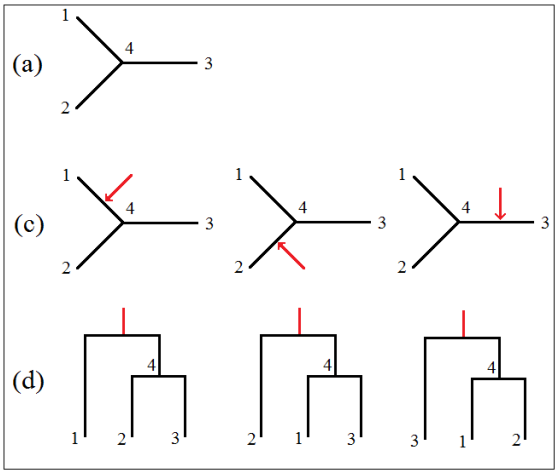


Fig. 12.3.3.c şi d. Arbori cu rădăcină derivați din arbori fără rădăcină

La modul general, dintr-un arbore fără rădăcină cu n frunze care are 2n-3 ramuri putem plasa rădăcina pe oricare din ramuri, deci vom avea relațiile:

$$N_{AR} = N_{AFR} \times (2n-3) = (2n-3)!! \qquad (12.3.3.b)$$

12.4. Construcția arborilor filogenetici - generalități

12.4.1. Date pentru construcția arborilor

În construcția arborilor filogenetici pornim de la o serie de date experimentale:

a) Distanțe – măsuri sau estimări ale distanțelor între specii sau între gene,

b) Caractere – colecția de aspecte morfologice (ex.: forma boabelor, culoarea ochilor, mărimea aripilor etc.), secvențe ADN sau proteine,

c) Ordinea genelor – din hărțile genetice după ordinea lineară a genelor ortoloage în genomurile date.

12.4.2. Metode de construcție a arborilor filogenetici

Au fost elaborate mai multe metode de construcție a arborilor filogenetici ce pot fi grupate în mai multe clase.

a) metode bazate pe distanțe – arborele explică distanțele evolutive estimate,

b) metoda parsimoniei – se alege arborele care necesită numărul minim de schimbări pentru a explica datele,

c) metoda asemănării maxime – pornind de la modele probabiliste de evoluție.

În cele ce urmează vom prezenta metodele bazate pe distanțe şi metoda parsimoniei.

12.4.3. Comparație între metodele bazate pe distanțe şi cele bazate pe parsimonie

A. Să considerăm secvențele frunzele de mai jos (cu câte 7 elemente, prezentate în tabelul 12.4.3.a).

Tabel 12.4.3.a. Exemplu pentru compararea metodelor de construcție a arborilor filogenetici

secvente \ pozitii	1	2	3	4	5	6	7
1	t	t	a	t	t	a	a
2	a	a	t	t	t	a	a
3	a	a	a	a	a	t	a
4	a	a	a	a	a	a	t

B. În metodele bazate pe distanțe construim întâi o matrice a distanțelor între toate perechile. Luăm cea mai simplă distanță – distanța Hamming, ce exprimă numărul de nepotriviri, prezentată în tabelul 12.4.3.b.

Tabel 12.4.3.b. Matricea distanțelor pentru secvențele din tabelul 12.4.3.a

Distanțe

secvente	1	2	3	4
1	0			
2	3	0		
3	5	4	0	
4	5	4	2	0
secvente	1	2	3	4

Fără a intra acum în detalii privind modul de construcție, prezentăm arborele din figura 12.4.3.a, care satisface distanțele între frunze – pe fiecare ramură a fost precizată „lungimea/ponderea" sa. Distanța între oricare două frunze este suma lungimilor ramurilor între ele.

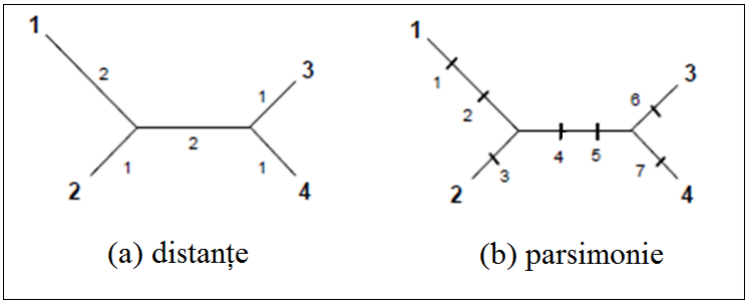


Fig. 12.4.3. Arborii corespunzători secvențelor din tabelul 12.4.3.a. Varianta (a) are pe ramuri lungimile corespunzătoare conform metodei distanțelor. Varianta (b) corespunde metodei parsimoniei, având pe ramuri precizate pozițiile care trebuie schimbate.

C. În metodele bazate pe parsimonie se precizează schimbările necesare pentru a alinia secvențele şi se alege arborele care necesită numărul minim de schimbări. În figura 12. 4. 3. b sunt prezentate chiar şi pozițiile din secvență care trebuie modificate la alinierea secvențelor. Numărul total de modificări între două secvențe este totalul de pe ramurile ce unesc cele două frunze.

12.5. Metode bazate pe distanțe

12.5.1. Proprietățile distanțelor, formularea problemei

A. Metodele bazate pe distanțe sunt simple şi au un caracter intuitiv. În capitolul 7 am descris o serie de abordări privind exprimarea „distanțelor" între diferite secvențe, distanțele fiind invers proporționale cu gradul de similaritate.

B. Proprietățile distanțelor

Din punct de vedere formal, orice metrică ce introduce o distanță între două secvențe, x_i şi x_j, notate d_{ij} sau dist (x_i, x_j), trebuie să acorde distanțelor următoarele proprietăți:

(a) $dist\ (x_i, x_j) \geq 0$ (distanțe pozitiv definite)
(b) $dist\ (x_i, x_j) = 0$ (distanța punctuală)
(c) $dist\ (x_i, x_j) = dist\ (x_j, x_i)$ (simetria)
(d) $dist\ (x_i, x_j) \leq dist\ (x_i, x_k) + dist\ (x_k, x_j)$ (regula triunghiului)

C. Fiind dată o matrice M între taxonii i şi j de dimensiune n × n (n = număr de taxoni /frunze), să se construiască un arbore cu ramuri ponderate ("edge weighted tree").

D. În figura 12.5.1.a este prezentat un exemplu de matrice a distanțelor iar alături este arborele corespunzător în varianta arbore cu rădăcină, având şi o scară pentru proporționalitatea lungimii ramurilor.

În cele ce urmează vom descrie un algoritm utilizat frecvent pentru construcția arborilor filogenetici prin metoda distanțelor.

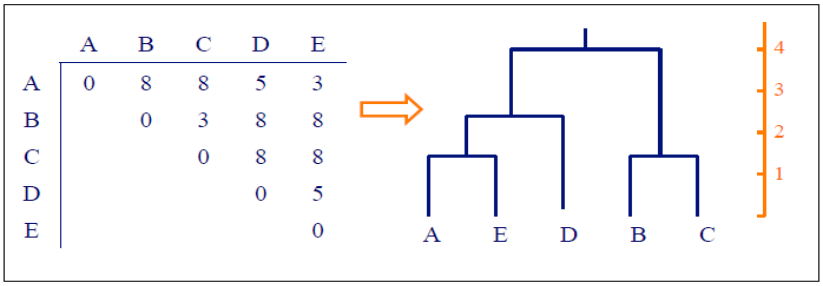


Fig. 12.5.1.a. Exemplu de arbore prin metoda distanțelor

12.5.2. Algoritmul UPGMA

A. Algoritmul UPGMA a fost propus de Sokal&Michner (1958) cu numele de "Unweighted Pair Group Method using arithmetic Averages". Este o metoda simplă şi intuitivă.

B. Ideea de bază este:
- compunerea succesivă a câte două componente (taxoni, clustere) formând un (nou) cluster
- se creează un nou nod pentru noul cluster

distanța între două clustere (între taxonii sau perechi de taxoni din fiecare cluster) este o distanță ponderată definită prin (12.5.2.a):

$$d_{ij} = \frac{1}{|C_i||C_j|} \sum_{p \in C_i, q \in C_j} d_{pq} \qquad (12.5.2.a)$$

C. Descrierea algoritmului UPGMA

- Se consideră fiecare taxon ca un cluster
- Se defineşte o frunză pentru fiecare taxon; se plasează la înălțimea "0" pe scara distanțelor
- Când sunt mai mult de două clustere:
  - Se aleg două clustere, ***i*** şi ***j***, pentru care distanța d_{ij} este minimă
  - Se defineşte un nou cluster $C_k = C_i \cup C_j$
  - Se defineşe un nod ***k*** părinte al ***i*** şi ***j***; se plasează la înălțimea $d_{ij} / 2$
  - Se înlocuiesc clusterele ***i*** şi ***j*** cu ***k***
  - Se calculează distanța între ***k*** şi celelalte clustere.

Ultimele două clustere ***i*** şi ***j*** se unesc cu o rădăcină la înălțimea $d_{ij} / 2$.

D. Să luăm un exemplu şi să construim arborele urmând pas cu pas algoritmul UPGMA.

	A	B	C	D
A	O	3	5	6
B		O	7	8
C			O	10
D				O

	C	D	E
C	O	10	6
D		O	7
E			O

	E	F
E	O	8
F		O

Clusterul c(E)=c(A)+c(B); c(A)=1, c(B)=1, c(E)=2

d(EC)=[d(AC).c(A)+d(BC).c(B)] / [c(A)+c(B)]

d(EC)=[5x1+7x1] / [1+1] = 12/2 = 6

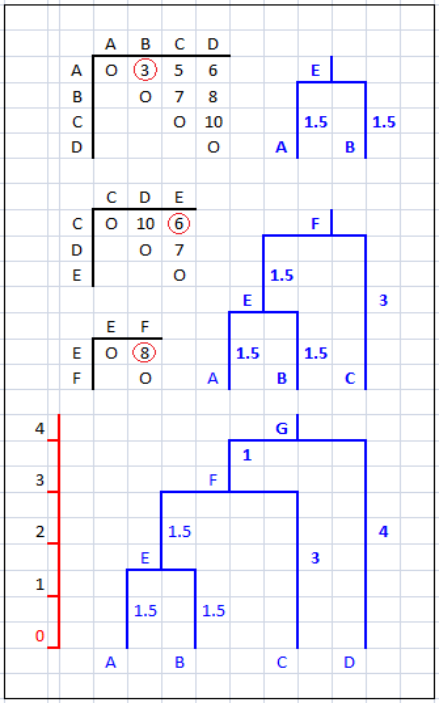
d(ED)=[d(AD).c(A)+d(BD).c(B)] / [c(A)+c(B)]

d(EC)=[6x1+8x1] / [1+1] = 14/2 = 7

Clusterul c(F)=c(C)+c(E)

d(FD)=[d(CD).c(C)+d(ED).c(E)] / [c(C)+c(E)]

d(FD)=[10x1+7x2] / [1+2] = 24/3 = 8



12.5.3. Distanțe ultrametrice

A. Algoritmul UPGMA descris mai sus presupune implicit o ipoteză numită „ipoteza ceasului molecular", prin care se consideră că divergența secvențelor apare cu aceeaşi rată în orice punct din arbore. Datele care respectă această proporționalitate a divergenței cu timpul se numesc „date ultrametrice", iar distanțele corespunzătoare sunt „distanțe ultrametrice".

B. Totuşi, ipoteza nu este în general valabilă. Procesul de selecție variază în diverse perioade de timp, variază cu organismul, cu genele unui organism sau cu regiunile unei gene. Putem astfel avea situații în care două secvențe apropiate ca grad de rudenie să prezinte distanțe mai mari decât distanțele reale față de secvențe mai îndepărtate filogenetic.

C. Cu alte cuvinte, putem avea situații în care prin metoda UPGMA să construim un arbore nereal. O astfel de situație este exemplificată în figura 12.5.3, în care în partea stângă - (a), este reprezentat arborele real, iar în dreapta - (b), o reconstrucție prin algoritmul UPGMA - reconstrucție de fapt greşită.

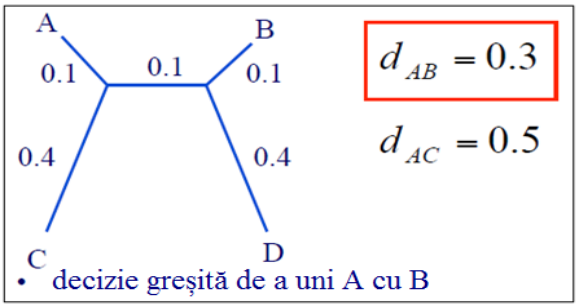


Fig. 12.5.3 Metoda Neighbor Joining

12.5.4. Metoda Neighbor Joining

A. Neajunsurile algoritmului UPGMA în cazul datelor care nu sunt ultrametrice, sunt înlăturate printr-o metodă derivată, numită "neighbor joining".

B. Într-un algoritm propus de Saitou & Nei [1987] în care distanțele reale d_{ij} între două frunze i şi j se corectează prin scăderea mediei distanțelor către toate celelalte frunze, ceea ce compensează ramurile lungi.

C. Din punct de vedere formal se definesc distanțele corectate D_{ij} prin relația

$$D_{ij} = d_{ij} - (r_i + r_j) \qquad (12.5.4.a)$$

unde media distanțelor de la frunza i la celelalte este

$$r_i = \frac{1}{|L|-2} \sum d_{ik} \qquad (12.5.4.b)$$

în care L este numărul de frunze.

În algoritmul Saitou se porneşte de la ideea că frunzele i şi j pentru care D_{ij} este minim sunt frunze vecine (adică provin din acelaşi nod proxim).

D. Algoritmul Saitou-Nei este oarecum asemănător cu UPGMA cu următoarele deosebiri:

- nu aplică ipoteza ceasului molecular
- se creează un arbore fără rădăcină
- presupune „aditivitate", adică distanța între perechile de frunze este suma lungimilor ramurilor care le conectează.

Nu prezentăm aici algoritmul în detaliu.

12.6. Metode bazate pe „parsimonie”

12.6.1. Enunțul problemei

A. Una dintre clasele de metode ce şi-a câştigat mulți adepți, atractivă prin simplitate şi logică, porneşte de la ipoteza că, pe linie filogenetică au apărut în mod natural secvențe divergente, însă nu cu multe modificări simultan, deci cea mai probabilă cale de căutare a unui posibil „părinte” este prin căutarea secvenței ce necesită un număr minim de schimbări.

Însăşi termenul are ca origine cuvântul „parcimonios”, cu semnificația de „econom, zgârcit”.

B. În metodele bazate pe „parsimonie” se caută arborele care explică datele cu un număr minim de schimbări.

Să subliniem faptul că nu mai urmărim lungimile ramurilor ci numai topologia.

C. Să luăm ca exemplu 4 secvențe simple AAG, AAA, GGA, AGA. Putem prezenta înrudirea lor şi evoluția către setul dat în mai multe variante, deci soluția nu este unică. Două din aceste soluții sunt prezentate în figura 12.6.1.

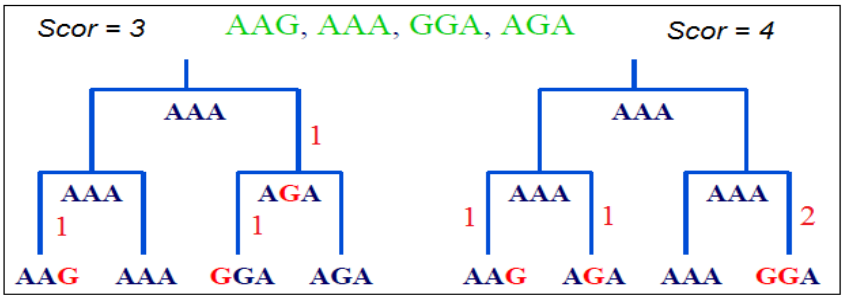


Fig. 12.6.1. Comparație între arbori pentru metoda parsimoniei

Se observă că în arborele din stânga evoluția a necesitat 3 schimbări (deci vom acorda scorul „3”), iar în arborele din dreapta scorul este mai slab, necesitând 4 schimbări.

D. Principiile algoritmilor din metoda parsimoniei

În algoritmii din clasa metodelor bazate pe parsimonie se pleacă de la următoarele ipoteze:

- orice element (stare, “state”) [nucleotid sau aminoacid] poate fi convertit în orice alt element,
- costurile schimbărilor sunt uniforme (Fitch) sau ponderate (Sankoff),
- pozițiile sunt independente (se poate calcula numărul minim separat pentru fiecare poziție).

12.6.2. Algoritmul lui Fitch

A. Cel mai popular algoritm este cel propus de Fitch (1971), în care arborele se traversează de două ori:

- în primul pas se traversează arborele de la frunze la rădăcină, determinând setul de stări posibile (de exemplu: nucleotide) pentru fiecare nod intern,

– în al doilea pas se traversează arborele de la rădăcină spre frunze alegând stările ancestrale pentru nodurile interne (un fel de ”trace-back”).

B. Vom descrie cei doi paşi, ilustrându-i şi cu un exemplu.

a) pasul 1: traversarea ”post-order” sau ”bottom-up”, adică de la frunze spre rădăcină.

Să stabilim stările posibile pentru nodurile interne.

– Se determină stările R_i ale nodului intern „i” care are copiii (urmaşii) „j” şi „k” conform regulii:
  - reuniunea R_j şi R_k dacă nu au elemente comune,
  - intersecția R_j cu R_k dacă au elemente comune.

Formal scriem

$$R_i = \begin{cases} R_j \cup R_k, & \text{if } R_j \cap R_k = \varnothing \\ R_j \cap R_k, & \text{otherwise} \end{cases} \qquad (12.6.2.a)$$

– Numărul de schimbări (scorul inițial) este dat de numărul de reuniuni!
– Avem în figura 12.6.2.a. un exemplu.

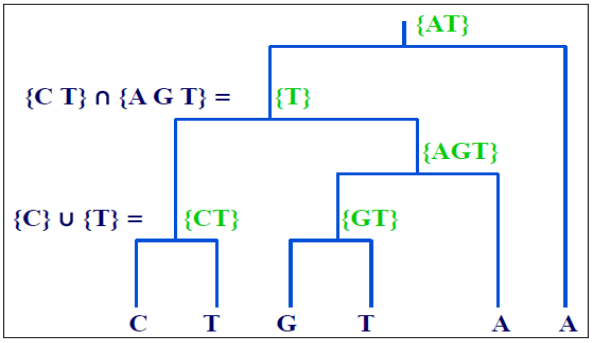


Fig. 12.6.2.a. Algoritmul Fitch, pasul „post-order”

Observăm că am efectuat 3 reuniuni şi două intersecții, deci scorul stabilit este s=3.

b) pasul 2: traversarea ”pre-order” sau ”top-down” de la rădăcină la frunze (echivalent „trace-back”).

Se alege starea r_j a unui nod intern „j” în funcție de starea R_i a părintelui „i” astfel:
  - egală cu cea a părintelui, dacă este inclusă
  - stare arbitrară în caz contrar

Formal scriem:

$$r_j = \begin{cases} r_i, \text{ if } r_i \in R_j \\ \text{arbitrary state} \in R_j, \text{ otherwise} \end{cases} \qquad (12.6.2.b)$$

Observație: în nod vom păstra starea după parcurgerea pasului 2.

Continuând exemplul de mai sus, marcăm cu literă închisă la culoare în figura 12.6.2.b starea din setul anterior care satisface relația (12.6.2.b.).

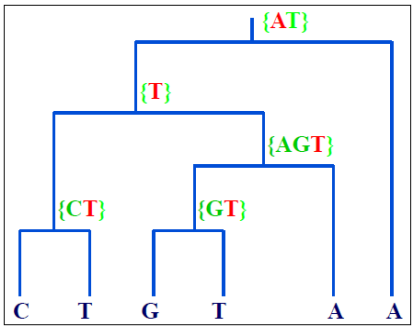


Fig. 12.6.2.b. Algoritmul Fitch, pasul „pre-order”

C. În final vom reține în noduri numai elementele selectate în pasul 2 şi vom marca cu „X” ramurile pe care s-a înregistrat substituție, pentru a evidenția scorul final (figura 12.6.2.c).

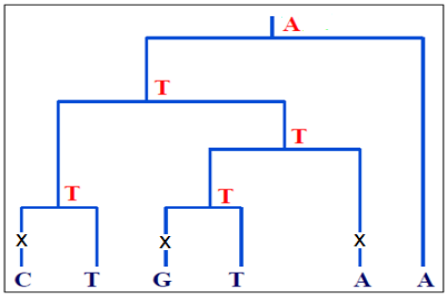


Fig. 12.6.2.c Arbore filogenetic construit prin metoda parsimoniei, algoritmul Fitch

12.6.3. Parsimonie ponderată

A. Varianta propusă de Sankoff & Cedergren [1983] aduce metodei parsimoniei aceleaşi ajustări ca modelul Kimura față de Jukes-Cantor, adică, în loc să considere toate schimbările echiprobabile, introduce costuri diferite S (a,b) pentru diverse substituții ale lui „a” cu „b” (tabelul 12.6.3).

B. Vom avea deci o matrice de substituție S, conținând costurile tuturor substitutelor posibile. De exemplu, în cazul secvențelor ADN vom acorda penalizări mai mici pentru tranziții (a↔g, c↔t) decât pentru transversii ({a,g}↔{c,t}). Conservarea va avea cost nul (nu se penalizează). În exemplul ce va fi prezentat mai jos apare o astfel de matrice.

Tabel 12.6.3. Matrice de substituție pentru metoda parsimoniei

	a	c	g	t
a	0	0.8	0.2	0.9
c	0.8	0	0.7	0.5
g	0.2	0.7	0	0.1
t	0.9	0.5	0.1	0

C. Primul pas al algoritmului Sankoff urmăreşte propagarea costurilor în sus pe arbore. Algoritmul propus este similar programării dinamice, având ca subproblemă să se determine costul $R_i(a)$ pentru subarborele cu rădăcina în „i”, plasând caracterul „a” în nodul „i”.

D. Formal vom putea descrie algoritmul prin următoarele relații, folosind notațiile din figura 12.6.3.a

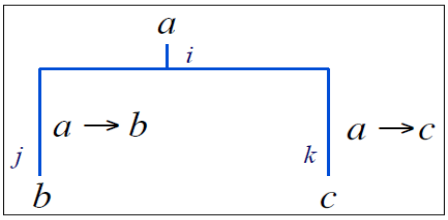


Fig. 12.6.3.a. Introducere de ponderi în parsimonia ponderată

- pentru frunze

$$R_i(a) = \begin{cases} 0 & daca\ "a"\ este\ un\ caracter\ in\ frunza \\ \infty & in\ caz\ contrar \end{cases} \quad (12.6.3.a)$$

- pentru noduri interne

$$R_i(a) = min_b\left[R_j(b) + s(a,b)\right] + min_c[R_k(c) + s(a,c)] \quad (12.6.3.b)$$

E. Să luăm un arbore (fig. 12.6.3.b) pe care să urmărim formulele costurilor pentru noduri.

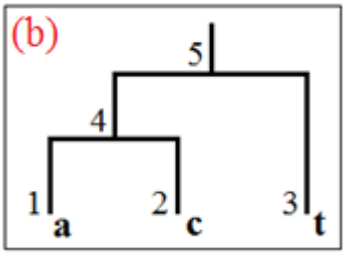


Fig. 12.6.3.b. Aplicație pentru parsimonia ponderată

Aplicăm relațiile (12.6.3.a) în cele 3 frunze, notate (1), (2) şi (3) apoi formulele (12.6.3.b) în nodurile (2) şi (1) şi obținem:

$R_1[a] = 0, R_1[c] = \infty, R_1[g] = \infty, R_1[t] = \infty$

$R_2[a] = \infty, R_2[c] = 0, R_2[g] = \infty, R_2[t] = \infty$

$R_3[a] = \infty, R_3[c] = \infty, R_3[g] = \infty, R_3[t] = 0$

$R_4[a] = \{R_1[a] + S(a,a)\} + \{R_2[c] + S(a,c)\} = \{0 + 0\} + \{0 + 0.8\} = 0.8$

$R_4[c] = \{R_1[a] + S(c,a)\} + \{R_2[c] + S(c,c)\} = \{0 + 0.8\} + \{0 + 0\} = 0.8$

$R_4[g] = \{R_1[a] + S(g,a)\} + \{R_2[c] + S(g,c)\} = \{0 + 0.2\} + \{0 + 0.7\} = 0.9$

$R_4[t] = \{R_1[a] + S(t,a)\} + \{R_2[c] + S(t,c)\} = \{0 + 0.9\} + \{0 + 0.5\} = 1.4$

$$R_5[a] = min\{R_4[a] + S(a,a), R_4[c] + S(a,c), R_4[g] + S(a,g), R_4[t] + S(a,t)\} + \{R_3[t] + S(a,t)\}$$
$$= \min\{0.8 + 0, 0.8 + 0.8, 0.9 + 0.2, 1.4 + 0.9\} + \{0 + 0.9\}$$
$$= min\{\mathbf{0.8}, 1.6, 1.1, 2.3\} + \{0.9\} = 0.8 + 0.9 = 1.7$$

$$R_5[c] = min\{R_4[a] + S(c,a), R_4[c] + S(c,c), R_4[g] + S(c,g), R_4[t] + S(c,t)\} + \{R_3[t] + S(c,t)\}$$
$$= \min\{0.8 + 0.8, 0.8 + 0, 0.9 + 0.7, 1.4 + 0.5\} + \{0 + 0.5\}$$
$$= min\{1.6, \mathbf{0.8}, 1.6, 1.9\} + \{0.5\} = 0.8 + 0.5 = 1.3$$

$$R_5[g] = min\{R_4[a] + S(g,a), R_4[c] + S(g,c), R_4[g] + S(g,g), R_4[t] + S(g,t)\} + \{R_3[t] + S(g,t)\}$$
$$= \min\{0.8 + 0.2, 0.8 + 0.7, 0.9 + 0, 1.4 + 0.1\} + \{0 + 0.1\}$$
$$= min\{1.0, 1.5, \mathbf{0.9}, 1.5\} + \{0.1\} = 0.9 + 0.1 = 1.0$$

$$R_5[t] = min\{R_4[a] + S(t,a), R_4[c] + S(t,c), R_4[g] + S(t,g), R_4[t] + S(t,t)\} + \{R_3[t] + S(t,t)\}$$
$$= \min\{0.8 + 0.9, 0.8 + 0.5, 0.9 + 0.1, 1.4 + 0\} + \{0 + 0\}$$
$$= min\{1.7, 1.3, \mathbf{1.0}, 1.4\} + \{0\} = 1.0 + 0 = 1.0$$

Sumar, pentru rădăcină:

$$R_5[a] = 1.7, R_5[c] = 1.3, R_5[g] = 1.0, R_5[t] = 1.0$$

F. Pasul 2: parcurgerea arborelui de sus în jos.

Observăm următoarele:

– caracterul de cost minim pentru nodul (5) este fie „g” fie „ t” cost (1. 0)
– caracterul de cost minim pentru nodul (4) este „ g”!

Acest lucru este impus de calea costului minim pentru nodul 1. Într-adevăr, observăm că minimul în $R_s[g]$ sau $R_s[t]$ se obține din termenul $R_4[g]$, chiar dacă valorile lui R_4 calculate din R_1 şi R_2 sunt mai mici pentru „a” sau „c”.

G. Rezultatul final: soluția este dublă, avem doi arbori cu cost minim prin metoda parsimoniei ponderate figura 12.6.3.c şi d.

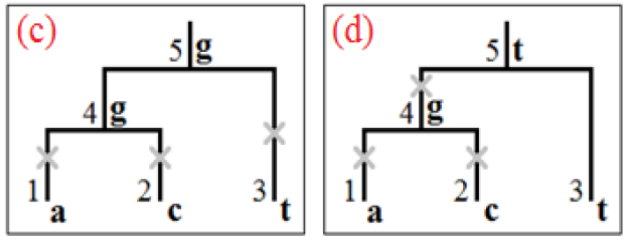


Fig. 12.6.3.c şi d. Arbori de cost minim prin metoda parsimoniei ponderate

Să verificăm costul în cele două cazuri:

- $S_{41} + S_{42} + S_{53} = S(g,a) + S(g,c) + S(g,t) = 0.2 + 0.7 + 0.1 = 1.0$
- $S_{41} + S_{42} + S_{54} = S(g,a) + S(g,c) + S(t,g) = 0.2 + 0.7 + 0.1 = 1.0$

Putem verifica şi că în (4) este mai potrivit să avem „g” decât „a” sau „c”, reprezentați în figura 12.6.3.e şi f.

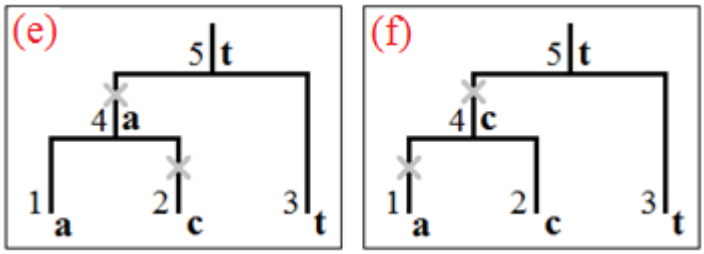


Fig. 12.6.3. e şi f. Calcul comparativ de costuri în parsimonia ponderată

Se observă costul mai ridicat, deşi ar fi fost efectuate doar două substituții în ambele cazuri!

La fel se poate arăta că plasarea lui „a” şi „c” în nodul 4 nu aduce cost minim nici în cazul în care în 5 luăm „g”.

12.7. Testarea arborilor - metoda „bootstrap”

A. Algoritmii de construcție a arborilor descrişi mai sus ne permit selectarea unui arbore „optim”, uneori, ca în cazul parsimoniei mai mulți, fără însă a da o măsură a calității arborelui ales. Desigur, este nevoie de o testare a semnificației unui aspect filogenetic, cum ar fi segregarea unui caracter pentru un grup de specii, etc.

B. Felsenstein [1985] a propus, o metodă, numită ”bootstrap”, dezvoltată ulterior de Efron & Tibshirani [1993], pe care o vom prezenta foarte pe scurt.

Fiind dat un set de secvențe aliniate, se generează un set artificial de aceeaşi dimensiune, preluând la întâmplare coloane din aliniere. Se aplică un algoritm de construcție a arborelui noului set de date. Se repetă procedeul de foarte multe ori (de ordinul miilor). Frecvența cu care apare un aspect filogenetic este luată ca măsură de încredere pe care o putem avea în aspectul respectiv.

Să menționăm în încheierea capitolului că pentru arborii filogenetici s-au dezvoltat şi metode probabiliste bazate în special pe metoda asemănării maxime.

Partea a II-a

1. BAZE DE DATE – BD integrată NCBI (I). Utilizarea motorului de căutare ENTREZ şi modulul PUBMED

1.1. Obiectivele lucrării de laborator

- căutări în bazele de date după cuvinte-cheie
- deprinderea creării interogărilor
- căutări de articole medicale în bazele de date după autor
- căutări de articole medicale în bazele de date după autor şi cuvinte-cheie.

1.2. Baze de date şi instrumente de căutare folosite în Bioinformatică

Bioinformatica tinde să dezvolte baze de date şi algoritmi din ce în ce mai puternici în scopul accelerării şi intensificării cercetării biologice. Disponibilitatea bazelor de date moleculare şi a instrumentelor bioinformatice au schimbat natura biologiei. Experimente care în trecut puteau fi efectuate doar într-un laborator, în urma efortului depus de-a lungul anilor, pot fi realizate cu ajutorul unui calculator personal şi o conexiune la internet, prin analiza şi interpretarea informației rezultate.

Pentru a îndeplini multitudinea de obiective la nivel global, a fost încurajată creşterea numărului bazelor de date din domeniul public. În prezent se poate obține o cantitate inimaginabilă de informații despre orice aspect al biologiei celulare prin intermediul Internetului, informații care pot fi bibliografice, genomice, structurale sau funcționale. Domeniul public, în afară de afişarea unor informații fundamentale, oferă utilitățile necesare analizei şi interpretării datelor. Aceste instrumente variază de la aliniamentul multiplu al secvențelor la studiile de expresie virtuală a genelor, PCR electronic şi altele. Câteva servere majore de biologie moleculară care funcționează în lume sunt:

- **NCBI (N**ational **C**entre for **B**iotechnology **I**nformation):
  http://www.ncbi.nlm.nih.gov
- **EBI (E**uropean **B**ioinformatics **I**nstitute):
  http://www.ebi.ac.uk
- **DDBJ (D**NA **D**ata**B**ank of **J**apan):
  http://www.ddbj.nig.ac.jp
- **PDB** (The **P**rotein **D**ata **B**ank):
  http://www.rcsb.org/pdb

Centrul Național de Informație Biotehnologică (NCBI) din Statele Unite ale Americii şi Institutul European de Bioinformatică (EBI) din Marea Britanie sunt principalele servere ştiințifice care mențin aceste imense baze de date precum şi instrumentele de software analitic necesare analizei datelor conținute.

Serviciile oferite de aceste servere sunt posibile datorită computerelor ultraperformante cu viteză de procesare mare care pot realiza prelucrările analitice ale datelor şi datorită Internetului care facilitează eforturile de comunicare electronică.

"Entrez" permite reconstituirea datelor din biologia moleculară şi vizualizarea citărilor bibliografice din bazele de date integrate ale NCBI (National Center for Biotechnology Information).

Acest motor de căutare se accesează la adresa: http://www.ncbi.nlm.nih.gov/Entrez/

Printre modulele incluse în baza de date integrată NCBI se găsesc:

- PubMed cuprinde citări şi articole biomedicale din BD Medline
- BD de secvenţe nucleotide (GenBank)
- BD de secvenţe proteice
- Structuri macromoleculare 3D.

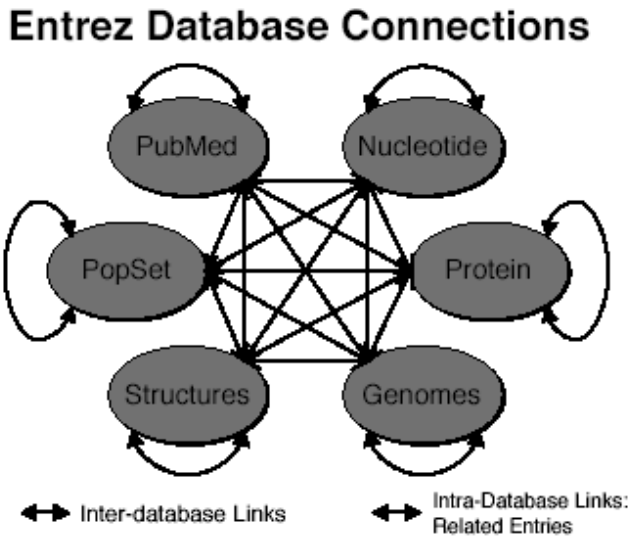


Figura 1.2.a. Legăturile intra şi inter baze de date

1.3. Utilizarea Entrez

Ne propunem să utilizăm motorul de căutare Entrez folosind următoarele exemple de căutări:

- **camp activated protein kinase** (figurile 1.3.a, 1.3.b)
- **"camp activated protein kinase"**
- **"camp activated" "protein kinase"**
- o **"camp dependent protein kinase"**
- o **immunoglob*.**

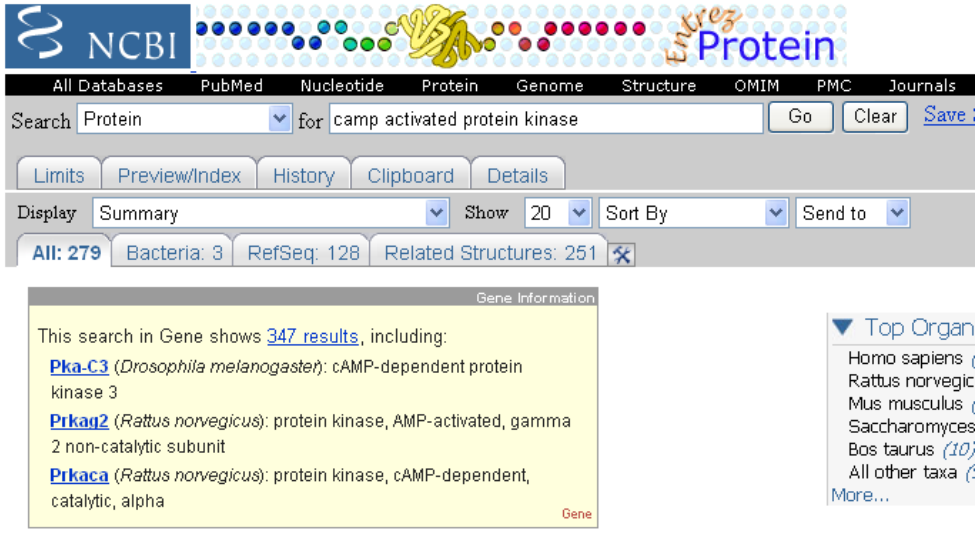


Figura 1.3.a. Interfața motorului de căutare

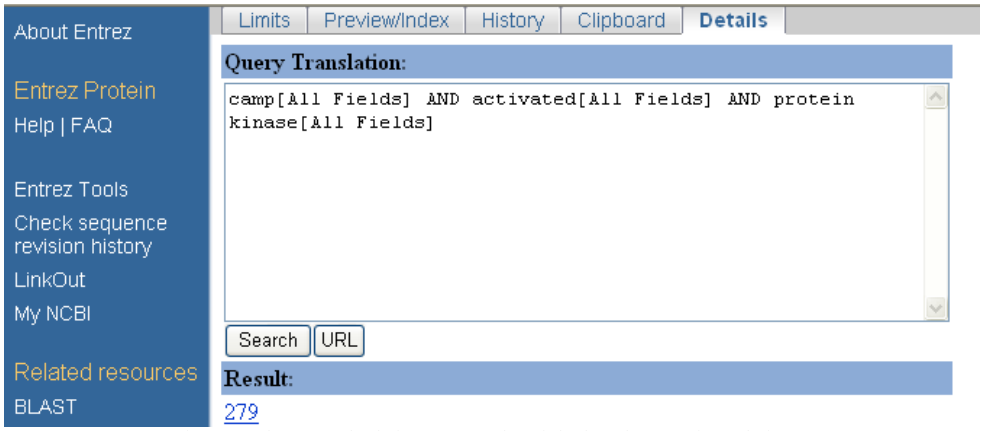


Figura 1.3.b. Exemplu de interogare a bazei de date după cuvinte-cheie

În acest scop urmăm paşii:

- se alege baza de date
- se introduce interogarea în fereastra "Details".

Intrați pe fiecare opțiune din fereastra ("Limits", "History" şi "Details") şi observați cum puteți aplica diverse limitări ale vizualizării informației rezultat, cum puteți opera cu interogările anterioare (figura 1.3.c.) şi care este expresia logică a interogării (figura 1.3.b).

Restul căutărilor le faceți individual şi observați diferențele ce survin în cadrul opțiunii "Details".

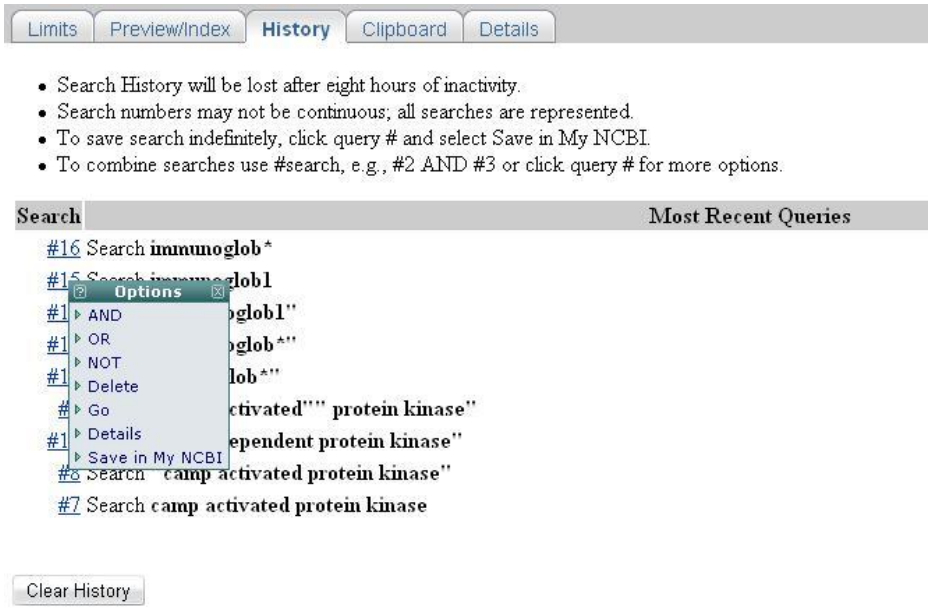


Figura 1.3.c. Prezentarea interogărilor anterioare

1.3.1. Calificatorii câmpului Entrez

Formatul general al calificatorilor "Entrez" sunt prezentați sub forma:

term [qualifier]

unde:

- **term** este şirul căutat
- **qualifier** reprezintă câmpul de căutat în BD specifică şi poate fi:
  - ACCN număr de acces
  - DATE anul publicației
  - JOUR numele jurnalului
  - MAJR topica MeSH
  - PAGE prima pagină
  - PROP proprietăți
  - PTYP tipul publicației
  - TITL cuvânt de titlu
  - SLEN lungimea secvenței
  - WORD cuvânt text
  - AUTH numele autorului
  - GENE numele genei
  - KYWD cuvânt cheie
  - MDAT data modificării
  - ORGN organismul
  - PDAT data publicării/creării
  - PROT numele proteinei
  - SQID id-ul secvenței
  - SUBS substanța
  - VOL volumul

1.3.2. Exemple

a. Căutaţi articole după **Bourne PE** ca autor.

Figura 1.3.2.a. Exemplu de căutare după **Bourne PE [AUTH]**

Dacă se doreşte o căutare avansată, formatul interogărilor booleene este:

term [field] operator **term [field]**

Operatorii pot fi:

– **AND** (intersecţie)
– **OR** (reuniune)
– **BUTNOT** (diferenţă)

b. Căutaţi articole ale autorului **Taylor P** care conţin informaţii despre **"protein kinase"** (figurile 1.3.2.b şi 1.3.2.c).

Figura 1.3.2.b. Exemplu de căutare după **Taylor P [AUTH] AND "protein kinase"**

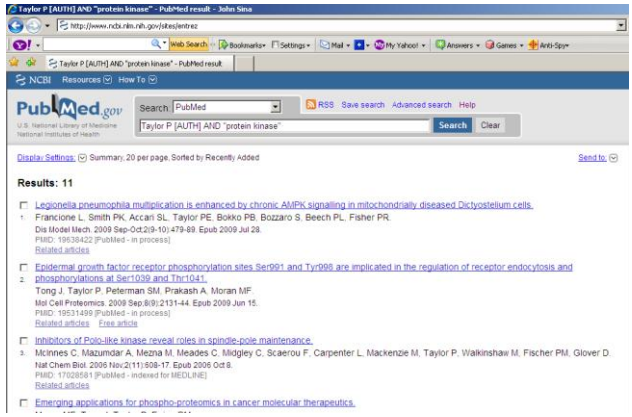


Figura 1.3.2.c. Exemplu de căutare cu ajutorul PubMed după **Taylor P [AUTH] AND "protein kinase"**

Notă:
Observați deosebirile între cele două motoare de căutare!

Rețineți că putem utiliza următoarele legături:
- PubMed Neighbours
- Text & MeSH
- Protein and Nucleotide Neighbours
- BLAST
- Macromolecular Structure Neighbours
- VAST

pe care le vom exemplifica în lucrarea de laborator următoare.

1.4. Utilizarea PubMed

Pentru a efectua căutări cu ajutorul secțiunii "PubMed" alegem opțiunea "**PubMed: biomedical literature citations and abstracts**" (figura 1.4.a).

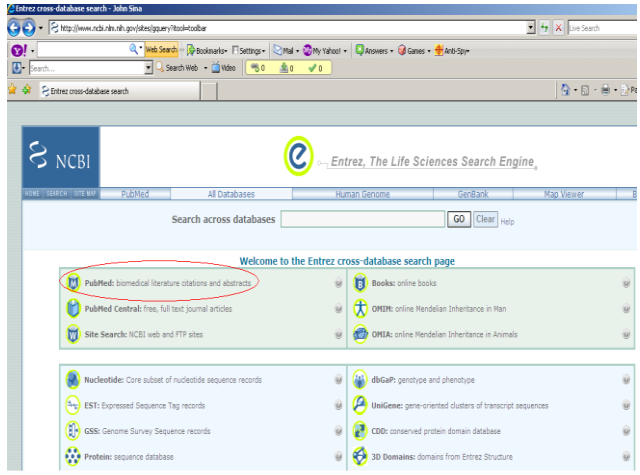


Figura 1.4.a. Opțiunea PubMed de căutare

Prin căutarea cu PubMed putem să:

- identificăm conceptele cheie din căutarea noastră
- introducem termenii în căsuța de search
- primim sugestii automate pe măsura completării termenilor în căsuța de căutare.

1.4.1. Exemple

a) Dacă dorim să aflăm ce rol are durerea în tulburările de somn introducem în căsuța de căutare cele două concepte-cheie: **pain** şi **sleeping disorders;**

b) Dacă dorim să căutăm după autor, introducem numele de familie al autorului plus inițialele, fără punctuație, în căsuță şi apăsăm butonul *search* (Ex. Watson JD, Lederberg J);

c) Apăsăm *Advanced search* pentru a căuta după *Autor, Jurnal, Data Publicării*. Pentru a căuta articole citate scrise de Bonnie W. Ramsey despre terapia genetică în fibroza chistică (cystic fibrosis), introducem următorii termeni de căutare în căsuță: **cystic fibrosis gene therapy ramsey bw**;

d) Publicații cu numele complet al autorilor pot fi căutate numai din 2002 (Ex. Joshua Lederberg, Garcia Algar, Oscar);

e) Dacă ştim numai numele de familie al autorului vom introduce în căsuța de căutare după autor urmatoarele: **numeautor[au];**

f) Pentru a căuta citate despre drosophila (musculița de oțet) în jurnalul "Molecular Biology of the Cell" introducem următoarele în căsuța de căutare: **molecular biology of the cell drosophila;**

g) Dacă căutăm studii sistematice despre terapia inhalatorie în pneumonie, din pagina "Clinical queries" apăsăm *Systematic Reviews* şi introducem următorii termeni în căsuța de căutare: **inhalation therapy pneumonia** (figura 1.4.1.a);

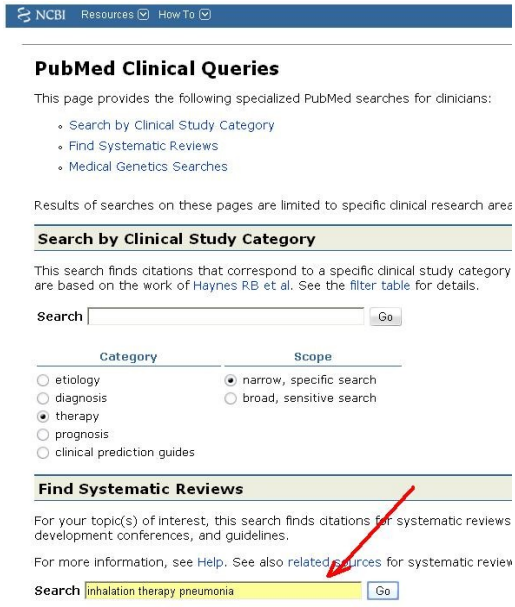


Figura 1.4.1.a. Exemplu de interogare clinică prin intermediul PubMed

h) Pentru a găsi informații despre anemia falciforma (siclemia) din "Clinical Queries" apăsăm *Medical Genetic Search*, apoi facem click pe *All check box* pentru a deselecta toate categoriile şi *Genetic Counseling*, după care introducem următorii termeni în căsuța de căutare: **sickle cell anemia.**

1.5. Exerciții propuse

– Lucrați cu tutorialul Entrez de la adresa:

http://www.ncbi.nlm.nih.gov/Database/tut1.html

– Găsiți referințe despre paralizia facială. Vizualizați înregistrările în formatul: rezumat şi autor.

– Găsiți secvența acidului nucleic murinic implicat în apoptoză şi cancer. Descrieți toate înregistrările pe o pagină web.

– Găsiți secvența proteică cauzatoare de SIDA apărută în revista Nature. Salvați strategia de căutare pentru o căutare ulterioară.

2. BAZE DE DATE – BD integrată NCBI (II). Programul BLAST

2.1. Obiectivele lucrării de laborator

- căutări în bazele de date după secvențe de gene
- determinarea cromozomului pe care se află o secvență de nucleotide
- căutări în bazele de date după secvențe de nucleotide
- determinarea secvențelor de nucleotide responsabile de apariția unor diagnostice
- căutare în BD după secvența de aminoacizi
- găsirea gradului de potrivire al secvenței de amioacizi introduse.

2.2. Introducere

Bioinformatica, ca ştiință a organizării şi analizei datelor biologice complexe (reprezentate de proteine şi secvențe de ADN), îmbină biologia moleculară şi genetica cu tehnologia de calcul pentru a înțelege rețeaua complexă de interacțiuni dintre componentele individuale ale celulei vii şi pentru a le integra în comportamentul întregului organism fiind utile în procesul de diagnosticare a bolilor şi în stabilirea de noi strategii terapeutice.

Utilitatea bioinformaticii se observă mai ales în Proiectul Genomului Uman, care a avut drept scop identificarea celor 30.000 de gene din ADN-ul uman.

Bioinformatica progresează cu o rată uimitoare, iar ariile de implicare majoră sunt achiziția de noi secvențe, încorporarea lor sub forma bazelor de date clasificate, integrarea informațiilor oferite de secvențe cu cele oferite de structuri, dezvoltarea instrumentelor pentru data mining şi dezvoltarea unei platforme comune pentru folosirea resurselor. În acest scop s-a creat ***I**nternational **N**ucleotide **S**equence **D**atabase **C**ollaboration* (**INSDC**), care include cele trei baze de date genomice majore din lume: **GenBank**, **EMBL** şi **DDBJ**.

Colaborarea dintre informaticieni, medici cercetători, biologi, matematicieni şi biochimişti le-a permis acestora studiul bazei moleculare a unei boli cu ajutorul instrumentelor matematice şi a tehnicii de calcul prin:

- analiza secvenței unei gene sau a produsului genei de interes;
- înțelegerea mai bună a organizării genelor analizate;
- predicția structurii moleculelor analizate (proteine).

Disponibilitatea informației genomice oferă bioinformaticianului un nou set de provocări. În prezent, ariile predominante ale analizei datelor bioinformatice includ:

- aliniamentul secvențelor
- predicția structurii proteinelor.

Studiile de aliniament a secvențelor sunt (în general) de două tipuri:

- de *aliniament al secvențelor în pereche* realizat cu ajutorul programului **BLAST**

- de *aliniament multiplu al secvenţelor* realizat cu programe de tipul **CLUSTAL.**

În ambele cazuri ideea este de a găsi similaritățile sau diferenţele dintre seturile de secvenţe. Analiza secvenţei reprezintă un instrument foarte important în studiul relaţiilor de evoluţie în genoame, duplicarea genelor, îmbinarea genelor etc.

Datele generate de Proiectul Genomul Uman sunt depozitate în bănci de date a genelor, care stochează secvenţe de ADN. În prezent sunt disponibile bănci de date pentru secvenţele şi structurile proteice. Una din operaţiile de bază din bioinformatică constă în căutarea similarităţii (omologiei) dintre un fragment de ADN nou secvenţializat şi secvenţe de ADN provenite de la diferite organisme. Găsirea unei potriviri apropiate permite predicţia tipului de proteină codată de noua secvenţă. Deşi nu este posibilă deocamdată predicţia completă a funcţiei sau structurii unei proteine *de novo* pornind de la secvenţa sa, pot fi trase nişte concluzii folositoare în legătură cu structura şi funcţia proteinei, în special prin compararea secvenţei proteinei cu structură şi funcţie necunoscută cu secvenţe proteice a căror structuri şi funcţii se cunosc. Prin compararea secvenţelor proteice echivalente de la diferite specii animale, se pot trage concluzii asupra evoluţiei acestor specii dintr-un strămoş comun.

2.3. Programul BLAST

Un program popular de comparare a secvenţelor de ADN este **BLAST** (***B**asic **L**ocal **A**lignment **S**earch **T**ool*). BLAST face parte dintr-un pachet de programe destinat căutării de secvenţe proteice, accesibil în diverse forme la diferiţi furnizori, sau prin intermediul NCBI, care mai oferă şi ***Entrez***, un instrument de meta-căutare care acoperă mare parte a bazelor de date de la NCBI, inclusiv cele care găzduiesc structuri tridimensionale a proteinelor, genoamele complete ale organismelor şi trimiteri la jurnale ştiinţifice care însoţesc intrările din bazele de date.

Asocierea dezvoltărilor tehnologiei de calcul şi moleculare deschide noi oportunităţi cercetărilor genetice. Folosirea combinată a informaţiei oferită de secvenţe, a instrumentelor de calcul, a bazelor de date şi a biologiei tradiţionale creşte speranţa înţelegerii funcţiei şi reglajelor tuturor genelor şi proteinelor, precum şi a descifrării funcţiilor celulei.

BLAST reprezintă instrumentul de căutare a aliniamentului local de bază, fiind un set de programe de căutare a similarităţilor, creat pentru identificarea clasificării şi a omologilor potenţiali pentru o secvenţă dată.

Pentru a înţelege mai bine programele BLAST, trebuie cunoscute aspectele de bază ale aliniamentelor secvenţelor. Acestea sunt folosite în special pentru găsirea potenţialilor omologi ce vor fi folosiţi ulterior pentru prezicerea posibilelor funcţii ale secvenţei necunoscute sau pentru modelarea structurii sale tridimensionale.

Aliniamentul global este cel mai bun aliniament, pe întreaga lungime a secvenţelor specificate. Introducerea spaţiilor (gaps) în secvenţele respective permite alinierea lor pe întreaga lungime. Principalul avantaj al aliniamentului global este optimizarea sa pentru secvenţele care au un grad înalt de similaritate, fiind astfel folositor în etapa de aliniere a secvenţelor din procesul de modelare a structurii tridimensionale (bazat pe secvenţele omologe cu structură tridimensională cunoscută).

Metodele de căutare ale aliniamentului local găsesc aliniamentul optim între subregiuni sau regiuni locale ale secvenţelor specificate. Aliniamentul local este cel mai

potrivit pentru secvenţe care au regiuni localizate de similarităţi. Un program de căutare a aliniamentului local este folosit de exemplu pentru găsirea motivelor, domeniilor şi altor unităţi repetitive din secvenţele respective, precum şi pentru găsirea secvenţelor similare pentru secvenţa necunoscută într-o bază de date. Pe scurt, un program de căutare al aliniamentului local este cel mai bine folosit pentru identificarea unor regiuni secvenţiale mai scurte, cu un grad foarte mare de similaritate.

Toţi algoritmii de comparare a secvenţelor se bazează pe anumite *scheme de calcul a scorului aliniamentului*. Scorul aliniamentului este suma scorurilor mai mici, atribuite pentru fiecare din perechile sale de aminoacizi sau nucleotide. Majoritatea acestor algoritmi folosesc o ***matrice de scor*** pentru calcularea unui scor total fiecărui aliniament.

Teoria statistică folosită în programele BLAST a fost creată de Samuel Karlin şi Steven Altschul.

Toate programele BLAST folosesc o matrice de substituţie, atât în etapa de scanare a bazelor de date cât şi în procesul de aliniere a secvenţelor.

Schemele de substituţie sunt considerate a fi cele mai bune metode de calcul al scorului aliniamentelor şi se bazează pe analiza frecvenţei cu care un aminoacid observat este înlocuit de un alt aminoacid în proteinele ale căror secvenţe sunt aliniate.

Criteriile care diferenţiază matricele de scor depind de tipul scorului pe care se bazează, astfel avem:

a) *Schemă a scorului bazată pe „**identitate**”*

Conform acestei scheme, perechile de aminoacizi identici sau nucleotide identice primesc un scor pozitiv, în timp ce perechile non-identice primesc scorul 0. În general scorul pozitiv atribuit perechilor identice este egal cu 1. Scorul identităţii globale este apoi convertit simplu (identitate procentuală).

Avantaje: această schemă de calcul este simplă şi non-heuristică. Este bună în cazul secvenţelor cu grad înalt de similaritate.

Dezavantaje: schema este în general inferioară acelora care încorporează cunoştinţele suplimentare, datorită în special inegalităţilor perechilor non-identice. De exemplu o pereche alanină-valină este mai acceptată din punct de vedere biologic decât o pereche alanină-acid aspartic. Această schemă este mai puţin efectivă în detectarea secvenţelor sau a regiunilor secvenţiale cu un grad redus de similaritate. Procentul identităţii raportat de acest aliniament nu este întotdeauna un indicator de acurateţe a gradului de omologie prezent, în special datorită dependenţei acestui scor de lungime a secvenţei.

*b) Schemă de calcul a scorului bazată pe „**similaritate chimică**”*

Această schemă a fost concepută pentru a depăşi limitările asociate cu schema bazată pe „identităţi” şi evaluează perechile de aminoacizi în funcţie de caracteristicile lor chimice şi structurale.

Schemele folosite de McLachlan şi Feng încorporează în calcularea scorului proprietăţile aminoacizilor cum ar fi polaritate, sarcină, mărime şi caracteristici structurale.

Avantaje: introduce proprietățile aminoacizilor în calcularea scorului, lucru important deoarece anumite mutații care realizează o schimbare drastică în caracteristicile AA implicați au un impact mult mai mare asupra funcțiilor proteinelor decât altele. Aceste mutații, de exemplu schimbarea unui aminoacid polar cu unul non-polar, alterează mult mai mult structura şi funcția proteinei respective decât o mutație implicând aminoacizi cu proprietăți similare.

Dezavantaje: mutațiile observate în natură nu sunt întotdeauna explicate prin schemele simple de calculare a scorului.

*c) Schema de calcul bazată pe „**codul genetic**”*

Această metodă ia în considerare numărul minim de schimbări de baze la nivel genomic, necesar pentru convertirea unui aminoacid în altul.

d) *Schema de calcul bazată pe „**mutații observate**”*

Această metodă de calcul a scorului unui aliniament, se bazează pe frecvența mutațiilor observate în secvențele aliniate.

Schemele bazate pe mutațiile observate reprezintă mai bine fenomenele naturale decât acelea care încearcă să explice relațiile dintre secvențe folosind matrice de calcul bazate pe similaritate chimică, identitate şi cod genetic.

Algoritmii de căutare a similarităților secvențelor aliniate se bazează pe cele 210 perechi posibile de aminoacizi care sunt reprezentate de o matrice 20x20 de calcul a scorului. Numărul total de perechi posibile de aminoacizi este egal cu 210, „alfabetul” proteinelor fiind alcătuit din 20 AA. Perechile de aminoacizi identici primesc cel mai înalt scor în matrice, urmate de perechile de aminoacizi care au un anumit grad de similaritate (de ex.: Leucină şi Izoleucină) şi în final de acei aminoacizi care nu prezintă similarități (de ex.: Leucină şi Arginină).

Programele BLAST folosesc un algoritm heuristic care identifică aliniamentele locale, găsind omologii cu secvențele cele mai apropiate, într-un timp eficient.

Serverul BLAST suportă o varietate de programe analitice care sunt fie accesate prin rețeaua Internet, fie instalate în rețele locale pentru a mări viteza de analiză. Programul BLAST bazal nu permite introducerea gap-urilor în aliniamentele sale ceea ce va reduce senzitivitatea căutării. Cu toate acestea, datele de ieşire din program oferă aliniamente regionale multiple, care pot fi folosite pentru a anticipa gap-urile din secvența de interes şi cea din baza de date. În continuare sunt enumerate programele BLAST şi utilizarea lor.

a) **BLASTp**: acest program permite utilizatorului să caute similaritățile dintre secvența unei proteine necunoscute şi secvențele proteinelor dintr-o bază de date.

b) **BLASTx**: permite compararea secvențelor traduse în aminoacizi ale nucleotidelor cu secvențele proteinelor din bazele de date.

Secvența nucleotidică de interes este tradusă inițial în toate cele 6 catene de citire **ORF** (**O**pen **R**eading **F**rame) posibile. Acest program este folositor în special pentru găsirea erorilor de secvențializare a nucleotidelor, prin compararea secvenței de nucleotide tradusă în aminoacizii săi proteici potențiali dintr-o bază de date cu secvențe proteice.

c) **BLASTn**: cu ajutorul acestui program se compară o secvență nucleotidică de interes cu secvențele din bazele de date nucleotidice.

d) **tBLASTn**: permite căutarea similarităților dintre o secvență proteică şi secvențele traduse (translatate) ale nucleotidelor dintr-o bază de date.

Secvențele nucleotidice dintr-o bază de date sunt traduse inițial în fiecare din cele 6 catene de citire posibile şi sunt apoi comparate cu secvența proteinei de interes. Acest program este util pentru găsirea erorilor de secvențializare în proteine prin compararea secvenței proteinei respective cu omologii săi potențiali obținuți prin traducerea secvențelor nucleotidice dintr-o bază de date.

e) **tBLASTx**: se compară cele 6 traduceri ale catenelor de citire ale secvenței nucleotidice chestionabile cu cele 6 catene de citire traduse ale secvențelor nucleotidice dintr-o bază.

Noul pachet de programe BLAST este menținut pe serverul BLAST 2.0 capabil să optimizeze viteza de procesare şi senzitivitatea metodelor, adăugând pe de altă parte noi capacități ce permit rularea noilor programe PSI-BLAST şi GAPPED-BLAST.

GAPPED BLAST – algoritmul Gapped-BLAST permite introducerea gap-urilor în aliniamentele obținute cu ajutorul programului BLAST simplu.

Introducerea gap-urilor (input) previne segmentarea regiunilor similare ale secvențelor.

Datele de intrare ale algoritmului heuristic permit reflectarea relațiilor biologice asociate aliniamentului, în special situsurile active şi situsurile de legătură care au tendințe să fie mai conservate de-a lungul evoluției. Introducerea gap-urilor previne scindarea acestor regiuni în fragmente de secvențe mai puțin semnificative.

PSI-BLAST (**p**osition – **s**pecific **i**terated BLAST) rulează inițial programul Gapped-BLAST şi foloseşte aliniamentul de ieşire din acesta ca input pentru PSI-BLAST. Programul construieşte o matrice de calculare a scorului care înlocuieşte secvența originală şi este folosită pentru găsirea profilelor (secvențelor omologe) în următoarele iterații de căutare în baza de date.

Utilizatorul ar trebui să efectueze următoarele etape generale pentru rularea cu succes a programelor BLAST:

- Secvența de interes trebuie introdusă în format corect (de exemplu formatul FASTA – similarul formatului BLAST, de pe serverul EBI);
- Secvența astfel formulată va fi apoi copiată în fereastra "input sequence" a interfeței programului BLAST;
- În funcție de tipul secvenței analizate se selectează programul BLAST potrivit (de exemplu BLASTp pentru secvențele de proteine);
- În final trebuie selectată baza de date corespunzătoare. De exemplu, dacă utilizatorul este interesat numai în găsirea secvențelor omologe cu structură cunoscută trebuie selectată o bază de date ce conține structuri tridimensionale, cum ar fi PDB. Secvența de interes este transmisă serverului BLAST, iar rezultatele căutării în baza de date sunt obținute fie prin e-mail, fie văzute interactiv pe interfața Internet a programului BLAST.

Valoarea aşteptată, **E**, din datele de ieşire ale programului BLAST reprezintă numărul de potriviri, „perechi" găsite aleatoriu într-o bază de date. O valoare E=0 semnifică faptul că pentru anumite baze de date probabilitatea de a găsi o pereche în

mod aleatoriu este 0. Această valoare descreşte exponenţial cu creşterea valorilor scorului S. O valoare E egală cu 1 indică probabilitatea ca identificarea similarităţilor secvenţei să fie aleatoare.

Formatul **FASTA** constă în reprezentarea fiecărui aminoacid din structura proteinei printr-un cod format dintr-un singur caracter. Codurile acceptate de programele **BLAST** sunt:

A	alanină	P	prolină
B	aspartat sau asparagină	Q	glutamină
C	cysteină	R	arginină
D	aspartat	S	serină
E	glutamat	T	threonină
F	fenilalanină	U	selenocisteină
G	glicină	V	valină
H	histidină	W	triptofan
I	izoleucină	Y	tirozină
K	lizină	Z	glutamat sau glutamină
L	leucină	X	orice aminoacid
M	methionină	*	oprirea translaţiei
N	asparagină	-	spaţiu de lungime nedeterminată

2.4. Exemple

a. Alegem opţiunea “Nucleotide: core subset of nucleotide sequence records” (figura 2.4.a).

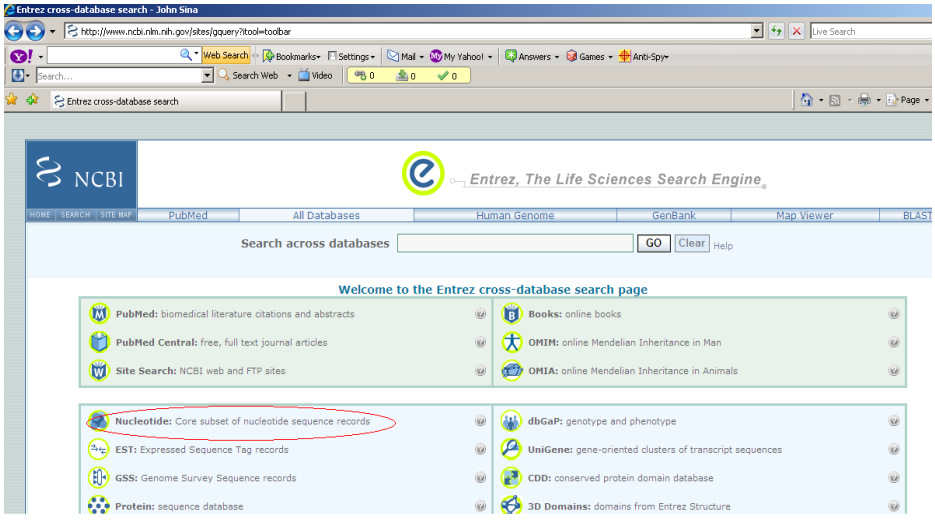


Figura 2.4.a. Interfaţa NCBI – opţiunea **Nucleotide**

Dacă dorim să căutăm secvenţa de gene care codează receptorul pentru endotelină (figura 2.4.b) introducem în căsuţa search următoarele: **Endothelin receptor.**

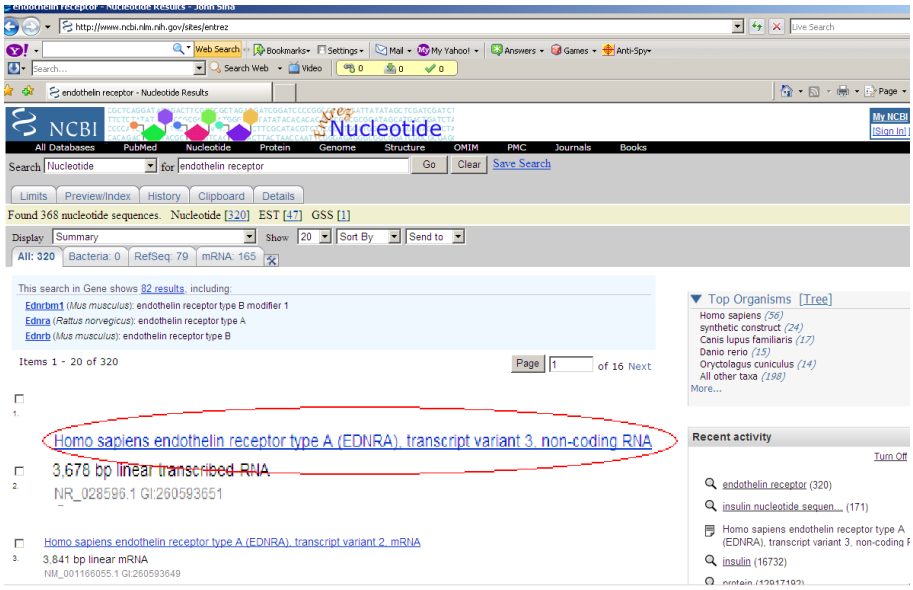


Figura 2.4.b. Exemplu de căutare a unei secvenţe de nucleotide

Din mulţimea de răspunsuri vom selecta varianta receptorului pentru specia umană (*homo sapiens*)!

Va fi afişat cromozomul pe care se află secvenţa de nucleotide (cromozomul 40) – figura 2.4.c, localizarea genei pe cromozom, secvenţa de nucleotide şi secvenţa codantă.

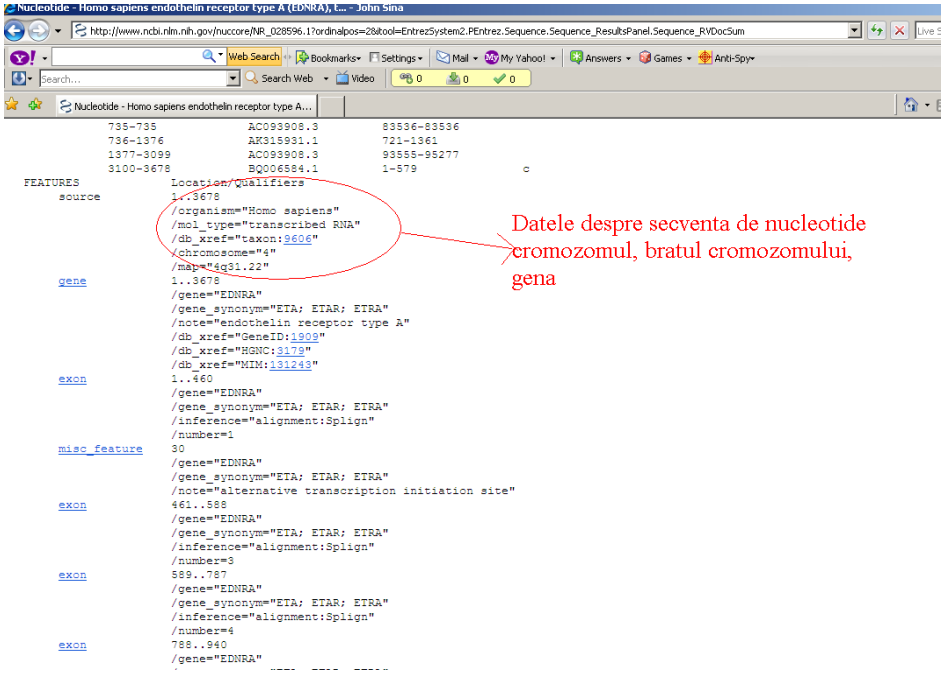


Figura 2.4.c. Informaţie despre secvenţa de nucleotide găsită

b. Pentru a căuta o secvență de nucleotide intrăm pe adresa: http://blast.ncbi.nlm.nih.gov/Blast.cgi (figura 2.4.d).

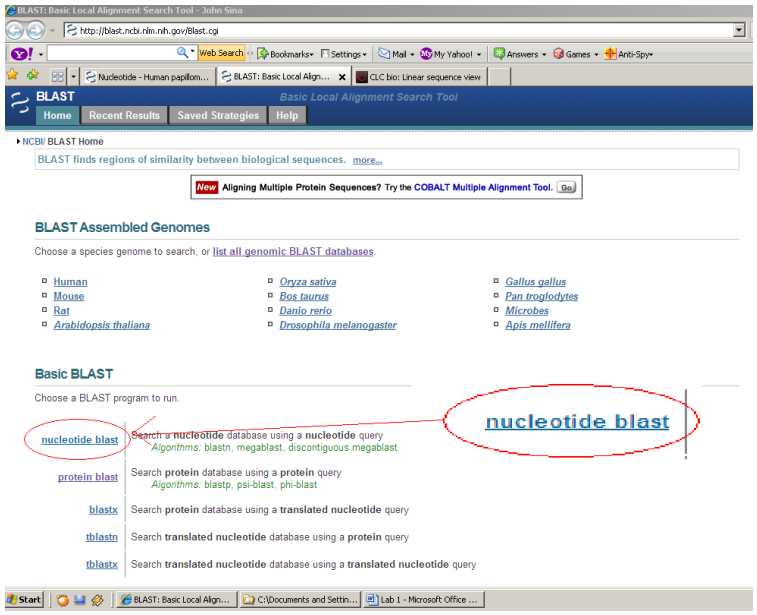


Figura 2.4.d. Opțiunile ferestrei de căutare BLAST

În căsuța de interogare (query) vom introduce următoarea secvență:

“1 ctagaaaccg tatgctatat aattatgtac tataaagtaa taatgtatac agtgtaatgg

61 atcatgggcc atgtgctttt caaactaatt gtacataaaa caagcatcta ttgaaaatat

121 ctgacaaact catcttttat ttttgatgtg tgtgtgtgtg tgtgtgtgtg ttttttaac

181 agggatttgg gg”

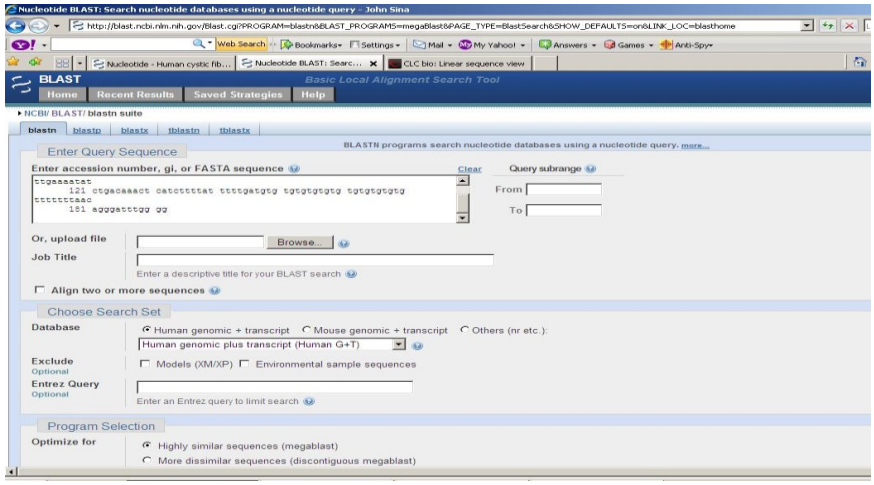


Figura 2.4.e. Interfața cu un exemplu de interogare BLASTn

Apoi vom da click pe butonul BLAST din josul paginii, ca în figura 2.4.f.

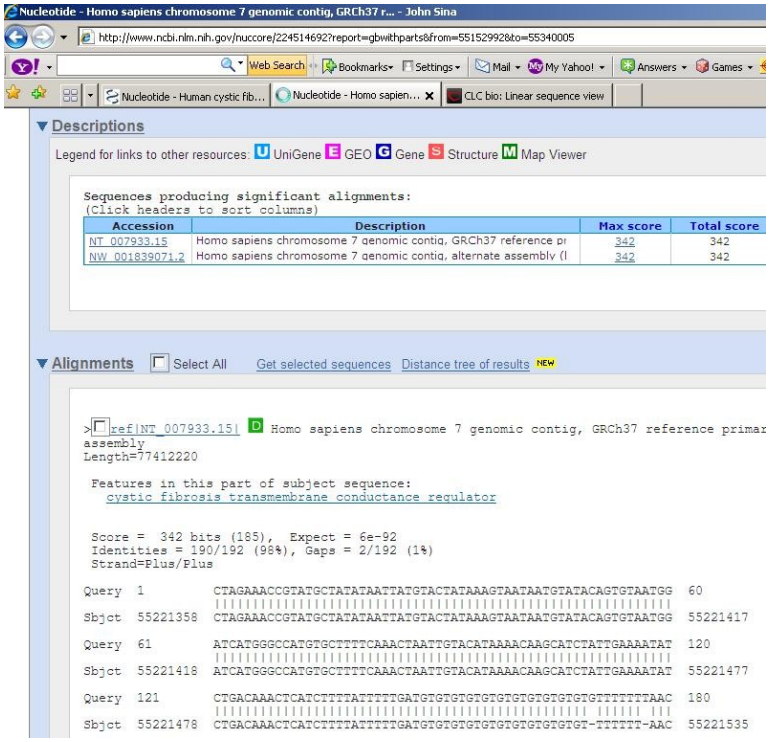


Figura 2.4.f. Rezultatul interogării

REZULTAT: Am găsit secvența de nucleotide responsabilă de apariția fibrozei chistice!

2.5. Utilizarea Bazelor de date cu secvențe proteice

a. Din pagina principală Entrez vom selecta "Protein: sequence database" (figura 2.5.a).

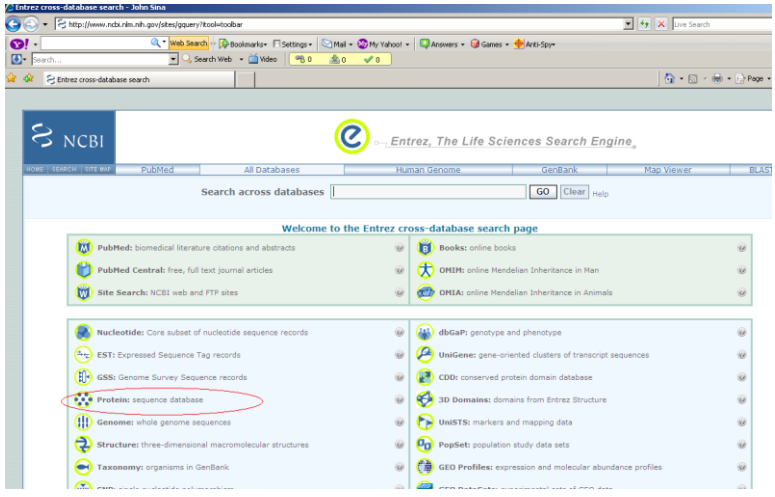


Figura 2.5.a. Opțiunea BD cu secvențe proteice

Dacă ne propunem să căutăm proteina insulină introducem în căsuța de search **insulin** (figura 2.5.b).

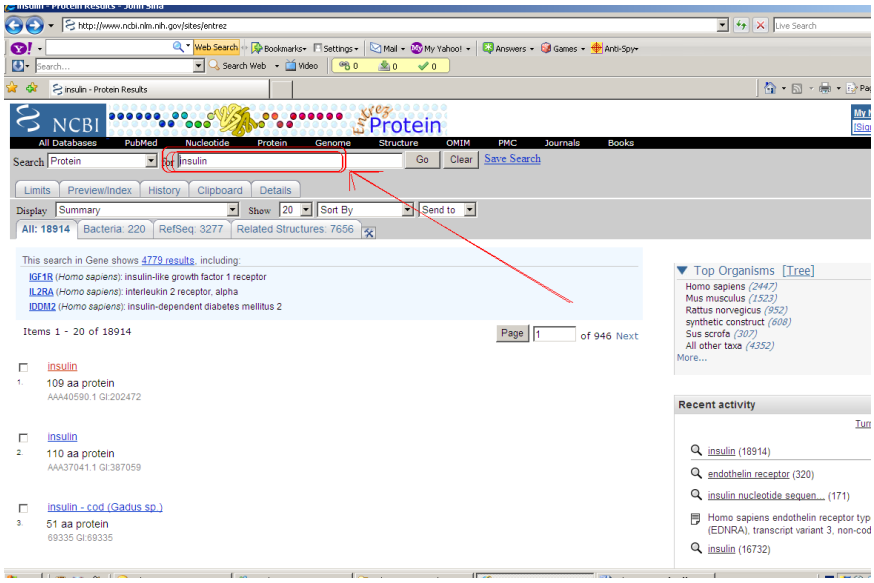


Figura 2.5.b. Exemplu de căutare a insulinei

Vom alege insulina corespunzătoare speciei umane (*homo sapiens*) şi vom găsi infomația potrivit căreia insulina umană este o proteină formată din 110 aminoacizi a căror reprezentare o puteți vedea în figura 2.5.c.

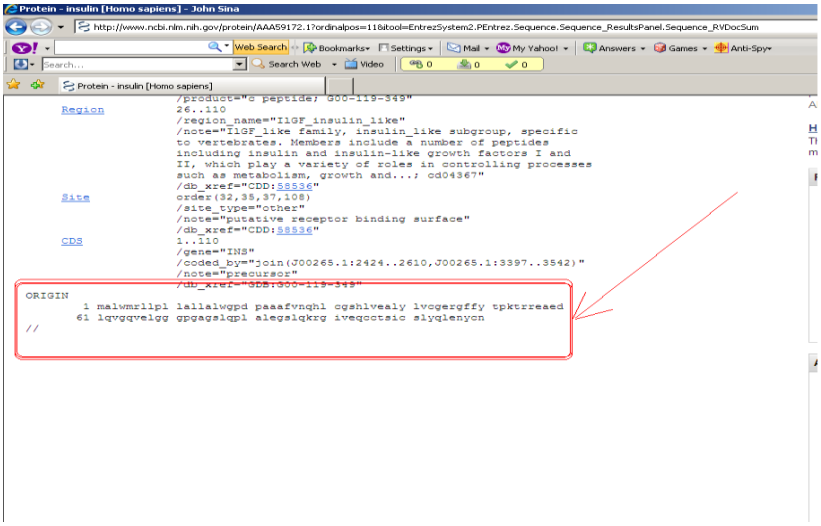


Figura 2.5.c. Secvența de aminoacizi ai proteinei insulina umană

În josul paginii va fi afişată secvența de aminoacizi din care e formată insulina, începând de la origine. După cum ştiți de la curs, fiecare aminoacid e codat cu o singură literă.

b. Introduceți secvența de aminoacizi **malwmrllpl** (figura 2.5.d).

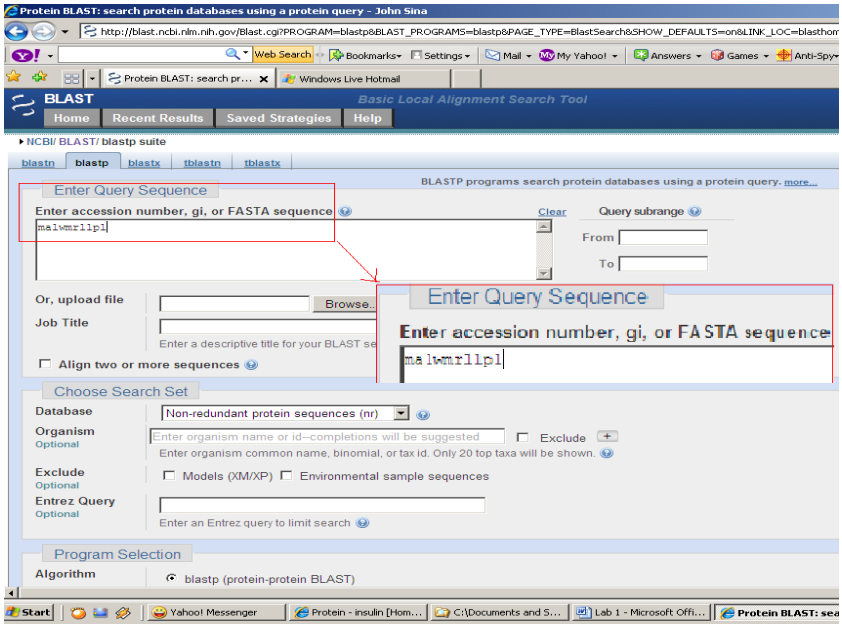


Figura 2.5.d. Interogarea BD după o secvență de aminoacizi

Apoi alegeți opțiunea cu butonul BLAST din josul paginii. Ca rezultat veți avea un grafic care va afișa gradul de potrivire al secvenței de amioacizi introdusă (figura 2.5.e). Sub grafic obțineți și rezultate text în ordinea potrivirii lor cu secvența introdusă inițial.

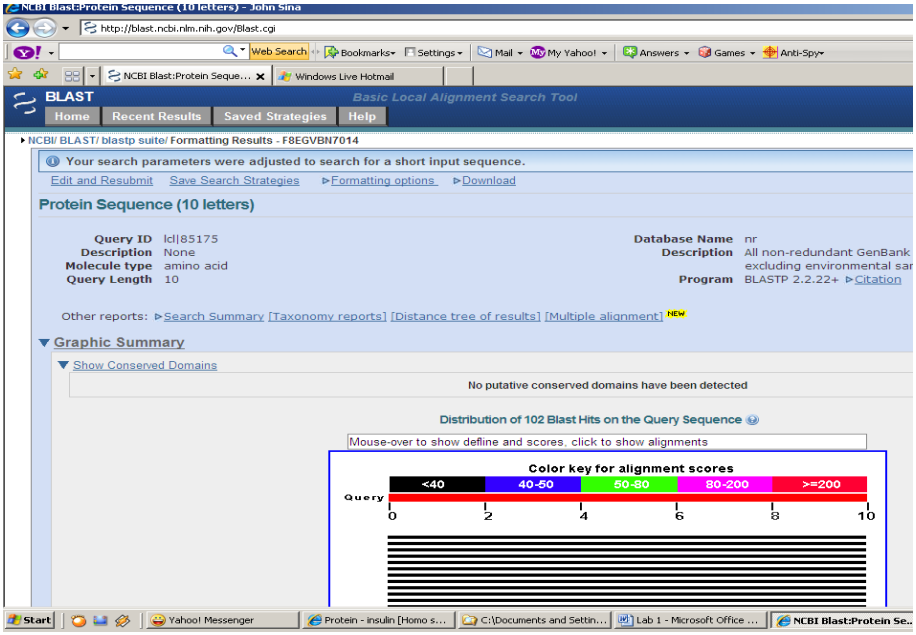


Figura 2.5.e. Rezultatul căutării gradului de potrivire al secvenței de aminoacizi

2.6. Exerciții propuse

Găsiți toate secvențele nucleotide de la şoarece şi om adăugate în BD Entrez în anul 1997. Câte structuri corespund rezultatului?

Vizualizați structura primară a hemoglobinei.

Copiați o secvență de aminoacid şi dați o căutare în BLAST (pentru a găsi asemănarea cu proteinele cunoscute).

Căutați în BLAST următoarea secvență de aminoacizi:
"1 mkwvtfisll flfssaysrg vfrrdahkse vahrfkdlge enfkalvlia faqylqqcpf
61 edhvklvnev tefaktcvad esaencdksl htlfgdklct vatlretyge madccakqep
121 ernecflqhk ddnpnlprlv rpevdvmcta fhdneetflk kylyeiarrh pyfyapellf
181 fakrykaaft eccqaadkaa cllpkldelr degkassakq rlkcaslqkf gerafkawav
241 arlsqrfpka efaevsklvt dltkvhtecc hgdllecadd radlakyice nqdsissklk
301 eccekpllek shciaevend empadlpsla adfveskdvc knyaeakdvf lgmflyeyar
361 rhpdysvvll lrlaktyett lekccaaadp hecyakvfde fkplveepqn likqncelfe
421 qlgeykfqna llvrytkkvp qvstptlvev srnlgkvgsk cckhpeakrm pcaedylsvv
481 lnqlcvlhek tpvsdrvtkc cteslvnrrp cfsalevdet yvpkefnaet ftfhadictl
541 sekerqikkq talvelvkhk pkatkeqlka vmddfaafve kcckaddket cfaeegkklv
601 aasqaalgl"

Căutați în BLAST următoarea secvență de nucleotide:
"1 gacacggctg tatatcactg tgcctctctt gacagcgcgt ttcgggggca ctggggccat
61 ggaaccctgg tcagcgtctc ctcagcatcc ccgaccagcc ccaaggtctt cccgctgagc
121 ctctgcagca cccagccaga tgggaacgtg gtcatcgcct gcctggtcca gggcttcttc
181 ccccaggagc cactcagtgt gacctggagc gaaagcggac agggcgtgac cgccagaaac
241 ttcccaccca gccaggatgc ctccggggac ctgtacacca cgagcagcca gctgaccctg
301 ccggccacac agtgcctagc cggcaagtcc gtgacatgcc acgtgaagca ctacacgaat"

Căutați secvența de nucleotide pentru gena RHD, RHCE (cromozomul 1). Căutați secvența de aminoacizi a imunoglobulinelor g (regiunea variabilă).

Căutați secvența de nucleotide care codifică sinteza fibrinogenului (factorul XIII al coagularii). Care este gena? (răspuns: F13A1).

Căutați următoarea secvență de aminoacizi:
1 mssvavltqe sfaehrsglv pqqikvatln seeesdppty kdafpplpek aaclesaqep
61 agawgnkirp ikasvitqvf hvpleerkyk dmnqfgegeq akicleimqr tgahlelsla
121 kdqglsimvs gkldavmkar kdivarlqtq asatvaipke hhrfvigkng eklqdlelkt

Căutați secvența de aminoacizi a mioglobinei.

3. Structura secundară, terțiară şi cuaternară a proteinelor. Programele PDB, Cn3D şi Rasmol

3.1. Obiectivele lucrării de laborator

- recunoaşterea structurii secundare, terțiare şi cuaternare a proteinelor
- exploatarea structurilor proteice cu PDB
- vizualizarea structurilor cu programul Cn3D
- vizualizarea structurilor cu programul Rasmol.

3.2. Noțiuni introductive

Proteinele sunt substanțe organice macromoleculare formate din lanțuri simple sau complexe de aminoacizi; ele sunt prezente în celulele tuturor organismelor vii în proporție de peste 50% din greutatea uscată. Toate proteinele sunt polimeri ai aminoacizilor în care secvența acestora este codificată de către o genă. Fiecare proteină are secvența ei unică de aminoacizi, determinată de secvența nucleotidică a genei.

3.2.1. Proprietăți fizico-chimice

a) Masa moleculară

Datorită formării aproape în exclusivitate din aminoacizi, putem considera proteinele ca fiind de fapt nişte polipeptide, cu masă moleculară foarte mare intre 10.000 şi 60.000.000. Masa moleculară se determină prin diferite metode, mai ales în cazul proteinelor cu masa moleculară foarte mare ca de exemplu proteina C reactiva. În următorul tabel sunt enumerate diferite proteine împreună cu masa moleculară corespunzătoare:

Tabel 3.2.1.a. Exemple de proteine, sursa şi masa moleculară

Denumirea proteinei	Sursa proteinei/Izolată din	Masa moleculară
Lactalbumină	lapte	17000
Gliadina	grâu	27.500
Insulina	pancreas	12.000
Hordeina	orz	27.500
Hemoglobina	globule roşii	68.000
Hemocianina	Moluşte (sânge), artropode (sânge)	2.800.00
Miozina	muşchi	850.000
Pepsină	stomac	36.000
Peroxidaza	rinichi	44.000
Virusul mozaicului tutunului (capsida)	tutun	17.000.000

Deoarece la multe proteine masa moleculară apare ca un multiplu de 17,500, multă vreme s-a mers pe ipoteza că particulele proteice sunt formate prin unirea mai multor molecule de bază ce au masa moleculară în jurul valorii de 17,500. Aceste molecule de bază s-ar putea uni între ele prin aşa numitele valenţe reziduale, ducând la formarea de *agregate moleculare*. Atunci când are loc ruperea acestor valenţe reziduale ar avea loc doar modificarea proprietăţilor fizice ale proteinelor, în timp ce dacă are loc ruperea legăturilor principale (legăturile peptidice), proteina îşi modifică proprietăţile fizico-chimice.

b) Solubilitatea proteinelor

Proteinele sunt substanţe solide, macromoleculare, solubile în general în apă şi insolubile în solvenţi organici nepolari. Unele proteine sunt solubile în apă dar insolubile în alcool, altele sunt solubile în soluţii apoase de electroliţi, acizi organici. Datorită gradului diferit de solubilitate în diferiţi solvenţi, proteinele se pot izola, identifica şi separa. Solubilitatea lor depinde foarte mult de legăturile care se stabilesc între grupările libere de la suprafaţa macromoleculelor şi moleculele solventului. La suprafaţa macromoleculelor proteice se găsesc grupări libere de tip polar:-COOH, -NH_2, -OH, -SH, -NH, grupări cu caracter hidrofil care favorizează dizolvarea proteinelor în apă. De asemenea există grupări de tip apolar, hidrofobe, de regulă radicali de hidrocarburi: -CH_3, -C_6H_5, -C_2H_5, care favorizează dizolvarea proteinelor în alcool. Însă în marea lor majoritate predomină grupările polare, determinante pentru caracterul hidrofil. În contact cu apa proteinele greu solubile manifestă fenomenul de gonflare, datorită tendinţei de hidratare datorată grupărilor polare. Gelatina de exemplu se îmbibă foarte puternic cu apa dând naştere prin răcire la geluri. La dizolvarea proteinelor în apă are loc fenomenul de formare a coloizilor hidrofili. S-a constatat că în soluţii diluate se găsesc macromolecule proteice izolate, iar în cazul soluţiilor concentrate se formează agregate de macromolecule proteice. Soluţiile coloidale ale proteinelor coagulează prin încălzire şi prezintă efectul Tyndall (dispersia fasciculului de lumină).

c) Structura secundară

Imaginea alfa helixurilor mioglobinei, a cărei structură a fost determinată de către Max Perutz şi Sir John Cowdery Kendrew în 1958 folosind cristalografia cu raze X, este prezentată în figura 3.2.1.a.

Figura 3.2.1.a. Structura secundară a mioglobinei

Structura secundară se referă la forma şi la lungimea lanţurilor polipeptidice, proprietăţi induse de legăturile de hidrogen. Cele mai întâlnite tipuri de structura

secundară sunt alpha helixul şi lanţurile beta (figura 3.2.1.b). Elicea alpha se formează prin rotaţia unui lanţ polipeptidic în jurul propriei axe.

Figura 3.2.1.b. Structura secundară sub formă de lanţuri beta

Alte helix-uri cum ar fi helixul 310 şi helixul π sunt din punct de vedere energetic favorabile formării legăturilor de hidrogen, dar sunt rareori observate în proteinele naturale exceptând părţile terminale ale helixului α în timpul formării scheletului proteic (de obicei centrul helixului). Aminoacizii au un comportament diferit vis-a-vis de posibilitatea formării structurii secundare. Prolina şi glicina sunt cunoscuţi ca aşa numiţii „helix breakers" (spărgători de helix), deoarece afectează configuraţia scheletului proteic; ambii aminoacizi au abilităţi conformaţionale neobişnuite şi de regulă se găsesc în colţurile scheletului proteic. Aminoacizii care preferă să adopte conformaţia helixului proteic fac parte din aşa numita serie MALEK (codurile formate din 1 literă a aminoacizilor: metionină, alanină, leucină, acid glutamic şi lizina); prin contrast aminoacizii aromatici (triptofanul, tirosina şi fenilalanina) dar şi aminoacizii cu legare prin carbonul beta (izoleucina, valina şi treonina) adoptă configuraţia β.

Structura secundară cunoaşte cîteva ipoteze privind formarea ei:

- Teoria polipeptidică formulată de către E. Hoffmeister în 1902 şi dezvoltată ulterior de către E. Fischer, are la bază conceptul conform căruia moleculele proteice sunt formate din lanţuri polipeptidice foarte lungi. Teoria are câteva dezavantaje:
  - nu explica diferenţierea biologică a anumitor proteine
  - unele proteine sunt rezistente la acţiunea enzimelor proteolitice (deşi datorită lungimii lanţului nu ar trebui).
- Teoria plierii şi răsucirii lanţului polipeptidice a fost elaborată de către Corey şi Pauling în 1943 şi a fost confirmată prin spectrele de difracţie cu raze X, microscopului electronic, prin măsurarea unghiurilor de valenţă, a distanţelor interatomice, au confirmat faptul că lanţul polipeptidic se găseşte sub formă pliată.
  - Plierea catenei are loc prin formarea legăturilor de hidrogen între gruparea carboxilică a unui aminoacid şi gruparea aminică a aminoacidului vecin. Lanţul polipetidic pliat se prezintă ca o panglică îndoită alternativ la dreapta şi la stânga, plierea având loc în dreptul carbonilor metinici. Mai multe lanţuri pliate polipeptidice dau naştere unei reţele, între aceste lanţuri pliate putându-se de asemenea forma legături de hidrogen, acestea fiind în număr mai mare când grupările terminale a două lanţuri sunt aranjate diferit (-NH_2 şi COOH, sau HOOC-şi -NH_2). Catenele polipeptidice pliate predomină în proteinele fibrilare şi mai puţin în cele globulare. După valoarea perioadei de identitate se cunosc mai multe tipuri de proteine cu structură pliată. Prin

perioada de identitate se înțelege distanța cea mai mică la care se repetă aminoacizii identici din moleculă.

- Structura α elicoidală, ipoteză lansată de Corey şi Pauling e o ipoteză conform căreia lanțul polipeptidic se poate prezenta şi înfăşura sub formă de spirală. În acest model, fiecare spiră conține de obicei 27 aminoacizi, iar distanța între spire este de 5,44 A^0. Fiecare aminoacid măreşte spira cu 1,47 A^0. În fața fiecărei grupări -CO- va apare la o distanță de 2,8A^0 o grupare NH de la al treilea aminoacid. Între aceste grupări se stabilesc punțile de hidrogen care asigură stabilitatea α helix-ului. În acest model lanțul polipeptidic se prezintă sub forma unui şurub cu pasul fie spre dreapta, fie spre stânga. În cazul proteinelor naturale, acestea datorită conținutului în L-aminoacizi, pasul helixului va fi spre dreapta, catenele laterale ies în afara corpului propriu-zis putând reacționa fie cu moleculele solventului fie cu alte catene polipeptidice. Canalul format în interiorul helixului este foarte îngust, în el nu poate pătrunde molecula solventului. Legăturile peptidice sunt plane, iar 2 planuri consecutive -CO-NH- formează un unghi de 180^0, rotirea lanțului făcându-se la carbonul α(metinic).

d) Structura terțiară

Prin intermediul cristalografiei cu raze X s-a dovedit faptul că macromoleculele proteice au o conformație tridimensională, realizată de obicei prin intermediul cuplării mai multor lanțuri polipeptidice scurte între ele, cuplare care duce la formarea fibrelor proteice; legăturile intercatenare pot fi principale sau secundare:

- Legături de hidrogen, sunt legături coordinativ heteropolare care se stabilesc cu uşurință între gruparea carbonil C=O (electronegativă) şi gruparea NH- (electropozitivă), din 2 lanțuri polipeptidice alăturate, sau în cazul formelor lactam-lactimă între gruparea -OH şi azotul iminic =NH (figura 3.2.1.c).

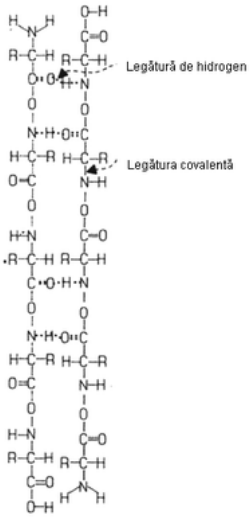


Figura 3.2.1.c. Exemplu de structură terțiară

Legăturile de hidrogen au lungimea cuprinsă între 2,7-3,1A şi energia de 3-7Kcal/mol la peptide, iar la apă 2-3Kcal/mol. Legăturile de hidrogen se pot stabili şi între catenele laterale care au grupări carboxil, hidroxil, amino sau tiolice. Din punct de

vedere energetic legătura de hidrogen nu este puternică dar datorită răspândirii relativ uniforme de-a lungul scheletului proteic oferă proteinei stabilitatea necesară.

– Legături disulfidice

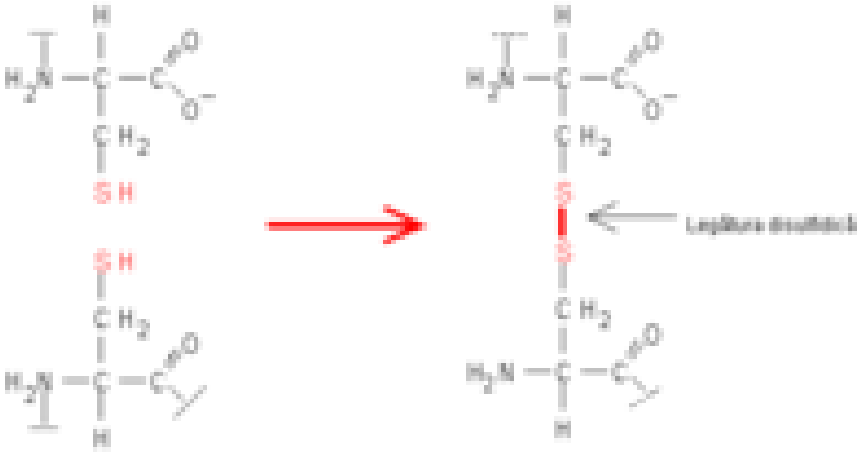


Figura 3.2.1.d. Exemplu de legătură disulfidică

Legătura disulfidică (figura 3.2.1.d) este foarte puternică, 50-100kcal/mol şi are un rol foarte important în stabilizarea arhitecturii spațiale a moleculei proteice. Legătura este rezistentă la hidroliză, însă se poate desface iar prin reducere formează tioli(SH), iar prin oxidare formează acizi. În general legătura sulfidică se întâlneşte la proteinele transformate care au o rezistență mecanică mare. În afară de aceste legături se mai pot stabili alte tipuri de legături: legături ionice (stabilite de obicei între grupările aminice şi cele carboxilice ionizate), legături de tip *van der Waals* (legături electrostatice slabe care se stabilesc între radicalii hidrofobi), legături fosfodiesterice (între două resturi de serină şi acid fosforic) şi legături eterice (stabilite la nivelul aminoacizilor cu grupări hidroxilice).

e) Structura cuaternară

Structura cuaternară se referă la modul cum se unesc subunitățile proteice. Enzimele care catalizează asamblarea acestor subunități poartă denumirea de holoenzime, în care o parte poartă denumirea de subunități reglatoare şi subunități catalitice.

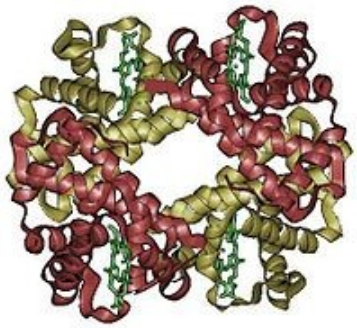


Figura 3.2.1.e. Vizualizarea 3D a hemoglobinei

Vederea 3 D a hemoglobinei (figura 3.2.1.e) prezintă 4 subunităţi roşu şi galben, iar unitatea hemică e verde. Numele de hemoglobină este format din hem şi globină, denumire ce denotă faptul că hemoglobina are la bază proteine globulare cuplate cu o grupare hem. Proteinele care au structura cuaternară sunt hemoglobina, ADN polimeraza şi canalele ionice, dar şi nucleozomi şi nanotubuli, care sunt complexe multiproteice. Fragmentele proteice pot suferi transformări în structura cuaternară, transformări care se reflectă fie în structurile individuale fie în reorientările fiecărei subunităţi proteice. Numerele subunităţilor din oligomerice sunt denumite prin adăugarea sufix-ului *-mer* (grecescul pentru subunitate), precedat de numele subunităţii, astfel:

Tabel 3.2.1.b. Numerotarea subunităţilor oligomerice

• 1 = monomer	• 7 = heptamer	• 13 = tridecamer	• 19 = nonadecamer
• 2 = dimer	• 8 = octamer	• 14 = tetradecamer	• 20 = eicosamer
• 3 = trimer	• 9 = nonamer	• 15 = pentadecamer*	• 21-mer
• 4 = tetramer	• 10 = decamer	• 16 = hexadecamer	• 22-mer
• 5 = pentamer	• 11 = undecamer	• 17 = heptadecamer*	• 23-mer
• 6 = hexamer	• 12 = dodecamer	• 18 = octadecamer	• etc.

3.3. Determinarea structurii proteinelor prin tehnici experimentale şi metode de modelare comparativă

Modelarea moleculară este ansamblu de activităţi întreprinse pentru calcularea unor proprietăţi moleculare, statice şi/sau dinamice, din care se pot estima proprietăţi microscopice. Modelarea moleculară caută să simuleze interacţiuni intra- şi inter-moleculare pentru a înţelege procesele fizico-chimice cu importanţă in biologie şi medicină, pentru a testa ipoteze şi pentru a prezice evenimente noi.

Un rol central în modelare îl joacă interacţia cu calculatorul, fie prin mijloace grafice fie prin tastatură-folosind un grup de comenzi care permit monitorizarea şi influenţarea stimulărilor.

Modelarea macromoleculelor biologice a fost iniţiată în anii 1940, cu mult înaintea apariţiei uneltelor computaţionale avansate. Modelarea structurilor macromoleculelor biologice ne permit studierea în profunzime a caracteristicilor funcţionale moleculare.

Primele macromolecule modelate au fost dublu-helixul de ADN şi mioglobina. Absenţa uneltelor computationale avansate a îngreunat mult aceste eforturi de modelare.

Linus Pauling a fost părintele modelării moleculare, structura alpha-helixului modelată de el permiţînd identificarea unor „modele" (pattern) structurale numite structuri secundare (alpha-helix, beta, strand) care există la nivelul molecular al proteinelor în toate organismele.

Sir Francisc Crick şi James Watson au primit Premiul Nobel pentru descoperirea structurii dublu catenare a moleculei de ADN, rezolvând astfel multe ambiguităţi legate de ereditate.

În zilele de astăzi, cristalografia, rezonanţa magnetică nucleară (NMR) şi microscopia crio-electronică ce permite defracţie electronică sunt tehnicile folosite pentru obţinerea structurilor de înaltă rezoluţie ale macromoleculelor biologice.

În cristalografie proteina cristalizată este bombardată cu electroni, modelul de difracție al electronilor fiind folosit pentru determinarea structurii atomice a moleculei.

Modelul de difracție este folosit pentru calcularea coordonatelor atomice bazat pe măsurarea densității electronice asociată cu fiecare dintre atomii detectați. Întrucât această metodă nu este limitată de mărimea moleculelor, poate fi folosită în principiu pentru determinarea structurii oricărei macromolecule şi în cristalografia cu dezavantaje: dificultatea de a obține cristale regulate a milioane de unități identice, multe din proteinele noastre esențiale (de exemplu proteinele membranare) sunt incapabile să cristalizeze când sunt îndepărtate din mediul lor; calcule extensive asociate cu analiza datelor. O altă limitare a metodei este lipsa informației legată de dinamica şi flexibilitatea macromoleculelor hologice în mediul lor, cristalografia oferind doar o imagine rapidă a unei molecule deformate expunerii de lungă durată necesară colectării modelelor de defracție.

O altă metodă folosită pentru determinarea structurii macromoleculare, este rezonanța magnetică nucleară. Chiar dacă metoda pare să fie avantajoasă şi rezonabilă, datorată folosirii semnăturilor protonice care, se suprapun în cazul moleculelor mai mari, impun folosirea NMR-ului în cazul macromoleculelor mai mici de 30 KD, de aceea sunt necesare tehnici alternative (custalograph, difracția electronică) pentru determinare structurii moleculelor mai mari.

Calcule extensive din analizele cristalografice şi RMN sunt efectuate cu acurateţe de programe automate şi semi automate ce reduc mai mult timpul necesar obținerii structurilor macromoleculare.

Cu ajutorul computerelor performante din ziua de azi, analiza unei molecule proteice tipică poate fi realizată în câteva zile în timp ce în absența acestor tehnologii ar fi necesari mult mai mulți ani pentru cercetarea respectivă. Cunoaşterea aspectelor structurale ale proteinei de interes va genera o imensă cantitate de informație despre funcția sa potențială şi despre relațiile cu alte macromolecule esențiale.

Predicția structurii de înaltă rezoluție a proteinelor şi al mecanismelor corecte de pliere al lor constituie însă o problemă majoră în biologie, deoarece proteinele sunt compuse din 20 de aminoacizi diferiți care introduc variabilitatea esențială a secvențelor observate la aceste molecule.

Fiecare din aceşti aminoacizi adopta variate conformații în raport cu celelalte reziduuri din proteină. Teoretic, numărul posibilităților structurale la care proteina se poate conforma în funcție de secvența sa de aminoacizi este fenomenal însă în realitate doar una sa câteva structuri active sunt formate în soluție.

Numeroase tehnici şi metodologii au fost utilizate pentru descoperirea acestor structuri active ale proteinelor din numărul imens de posibilități conformaționale, cele mai folosite fiind cele care folosesc relația proteine cu omologi cunoscuți şi cele de minimizare energetică axate pe termodinamică.

Aşa cum am amintit anterior, rata de obținere a structurilor macromoleculelor prin cristalografie sau rezonanță magnetică nucleară este mult mai mică decât numărul foarte mare de secvențe noi ADN şi proteice introdus zilnic, fiind astfel necesară o abordare automatizata a designului structural. Grupurile de informaticieni şi cercetători îşi direcționează eforturile spre identificarea macromoleculelor biologice-cheie (de ex. proteina) responsabile de procesele patologice şi propunerea inhibitorilor potențiali ai acestor molecule. Cel mai mare obstacol întâmpinat este lipsa datelor structurale ele

moleculei de interes. În majoritatea cazurilor macromoleculele studiate sunt proteine structurale sau reglatorii. În cazul moleculelor pentru care nu există date structurale obişnuite cristalografic sau prin NMR, pot fi folosite date ale proteinelor omologe care au o structura cunoscută. Modelarea comparativă este bazată pe similaritățile structurale dintre proteina necunoscută şi omologă sau cu structura tridimensională cunoscută şi unul din programele performante care realizează această modelare este programul „Homology" din pachetul software Insight II al companiei MSI.

În continuare vor fi prezentate detalii legate de modelarea comparativă cu avantajele şi limbajele acesteia.

Multe dintre proteinele esențiale prezente în organismul uman se găsesc şi în alte organisme vii. Aceste proteine au un rol cheie în menținerea vieții. De exemplu proteinele implicate în catalizarea unor procese esențiale, cum ar fi replicarea ADN-ului, trebuie sa-şi conserve funcționalitate de-a lungul evoluției şi sunt comune tuturor organismelor în timp ce alte proteine mai puțin esențiale sunt mai caracteristice pentru anumite organisme.

Proteinele omologe sunt de obicei proteine care au o compoziție similară de aminoacizi şi o origine evoluționară comună. Schimbări în secvența aminoacizilor din proteină poate conduce la schimbarea structurii sale tridimensionale. Această relație între secvența aminoacizilor din proteină şi structura tridimensională a proteinei ne permite compararea proteinelor care nu au date structurale oferite de cristalografie sau NMR cu secvențele omologe având structuri tridimensionale cunoscute.

În modelarea proteică bazată pe omologie, structurile determinate experimental reprezintă modelele „template" pentru secvențele omologe „țintă" care nu au coordonate structurale.

Protocolul standard al modelării comparative cuprinde 5 etape (figura 3.3.a):

1. izolarea secvenței ADN
2. găsirea omologilor cu structura cunoscută pentru proteina „țintă", această căutare putând fi realizată cu programul BLAST;
3. alinierea secvențelor „țintă" cu aceste secvențe „model" pentru identificarea structurii model cea mai apropiată;
4. construirea modelului tridimensional al secvenței „țintă" pornind de la modelele cele mai corespunzătoare găsite în etapa a 2-a;
5. evaluarea calitativă a modelului secvenței țintă, folosind criterii diverse (de ex. energetice, stereochimice).

În mod ideal compoziția în aminoacizi a unei proteine necunoscute şi cea cunoscută a omologilor săi structurali sunt oarecum similare şi în baza de date proteică (PDB) sunt prezente mai multe structuri omologe provenind din specii diferite.

Programul de căutare BLAST foloseşte ca şi date de intrare pentru căutare secvenței de aminoacizi a proteinei țintă, pentru căutarea secvențelor aminoacizilor omologe în baza de date specializată (de ex.: PDB, SWISS-Prot). La sfârşitul căutării sunt afişate structurile omologe cele mai apropiate de structura țintă cu atât mai înrudite cu cât scorul BLAST este mai mare.

Coordonatele structurale ale proteinelor celor mai similare cu proteina țintă sunt salvate şi folosite ca date de intrare pentru programul de modelare comparativă (de ex. software-ul „Homology" din Insight II).

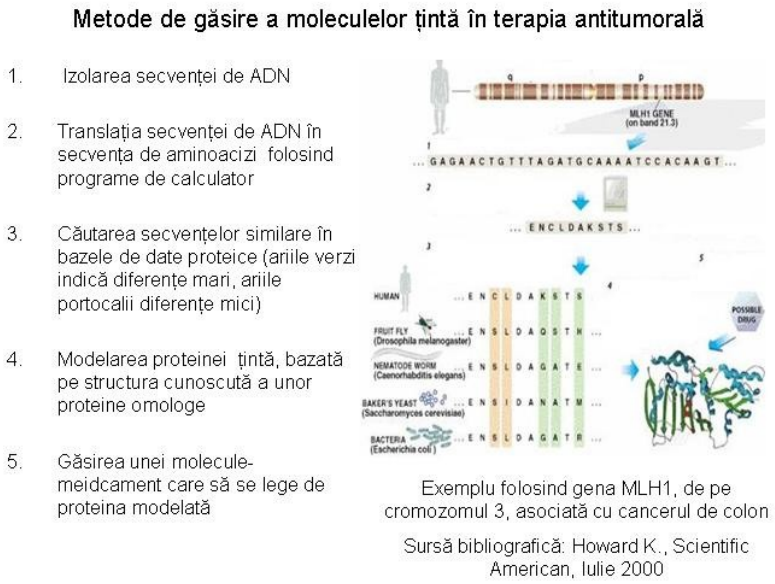


Figura 3.3.a. Protocolul standard al modelării comparative

Coordonatele atomice ale structurilor „matrița" (template) sunt aliniate structural pentru a reprezenta regiunile structurale conservate (RSC_S) ale proteinei „țintă". Folosirea SCR_S – urilor oferă posibilitatea construirii caracteristicilor structurale conservate evoluționist pentru proteina țintă.

Multe regiuni de buclă ale proteinelor omologe au o mare similaritate a secvențelor de aminoacizi dar au caracteristici structurale tridimensionale variabile. Cu alte cuvinte, relația între regiunile structurale conservate şi secvențele de aminoacizi aliniate nu este aşa de evidentă cum ar fi de aşteptat. De aceea în procesul modelării comparative este foarte importantă stabilirea acestor RSC_S, majoritatea acestei regiuni conservate fiind reprezentate de structurile secundare (α - helix, structură β, etc.). Structurile regiunilor din afara SCR-urilor sunt de obicei buclele structurale (loops) şi capetele terminale ale moleculelor.

După stabilirea coordonatelor aminoacizilor din secvențele suprapuse la nivelul SCR-urilor trebuie găsite cele mai corespunzătoare coordonate pentru regiunile de buclă ale proteinei țintă, precum şi ale capetelor terminale. Aceste deziderate pot fi realizate cu ajutorul programului „Homology" din pachetul de programe comerciale „Insight II" al companiei MSI.

Uneltele software de predicție a structurilor proteice sunt în general clasificate în două domenii: domeniul public accesibil şi cel comercial, privat.

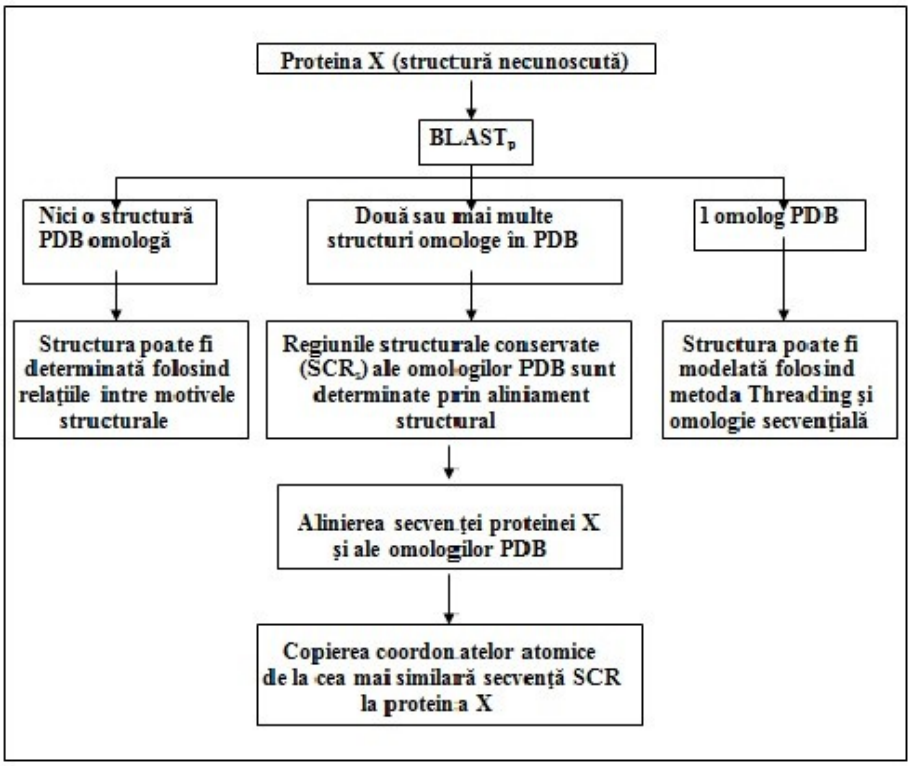


Figura 3.3.b. Modelarea structurilor proteice bazată pe omologie

Unul din programele comerciale foarte des folosit este mediul Insight II al companiei biotehnologiei şi bioinformaticii MSI (Molecular Simulation Inc.) care a revoluționat studiile de design medicamentos prin adăugarea elementelor de predicție a structurii proteice la abordarea empirică învechită.

La ora actuală programele Insight II rulează numai pe computere Silicon Graphics şi IBM RISC system/6000 workstation. Majoritatea modulelor software ale companiei MSI necesită programul grafic Insight II 3-D graphics, un mediu prietenos şi uşor de folosit de către utilizatori.

3.4. Grafica moleculară

Reprezentarea computerizată a modelelor biologice a fost realizată pentru prima oară de către Cyrus Levinthal şi colegii săi care au creat un sistem ce a afişat pe un osciloscop reprezentări ale structurilor macromoleculare.

Primul sistem pentru afişajul interactiv al structurilor moleculare a fost creat la MIT în 1966 (Eric Francoeur, „Early Interactive Molecules Graphics at MIT, 1998, http://www.umas.edu/molvis/francoeur/levinthal/lev-index.html).

Folosind unul din primele calculatoare, Project MAC, C. Levinthal şi colegii săi au conceput un program „model Building" care să lucreze cu structurile proteice.

Programul a permis studiul interacțiunilor de scurtă durată între atomi precum şi „manipularea" structurilor moleculare. Terminalul de afişare a structurii moleculare a

fost un osciloscop monocromic arătând structura în formă atomică cu reprezentare grafică de „sârmă". Efectul tridimensional a fost obținut prin rotația constantă a structurii pe ecran.

La începutul anilor 1970, pentru prima dată, structura unei proteine a fost obținută cristalografic şi vizualizată în întregime cu ajutorul calculatoarelor de către Jane şi David Richardson şi colegii. La sfârşitul anilor 1970, Thomas Porter a dezvoltat algoritmi pentru reprezentările spațiale care au revoluționat vizualizarea macromoleculelor dar au avut acces limitat, numai pentru specialişti.

În anii 1980 David şi Jane Richardson au realizat reprezentări grafice ale structurilor moleculare folosind programul CHAOS iar în 1992 au descris „Kinemage" (imaginea cinetică) şi programele create de ei, MAGE şi PREKIN.

- MAGE este accesibil la adresa http://kinemage.biochem.duke.edu,
„The Richardsons 3D Protein Structure Laboratory and Kinemage Home Page". Cuprinde următoarele:
- MAGE– pentru afisajul Kinemage-lor
- PREKIN prepară imaginile cinetice (Kinemage's) moleculare necesare ca input pentru programul MAGE, pornind de la coordonatele în format PDB
- REDUCE – pentru adăugarea şi optimizarea hidrogenilor
- PROBE – calculul contactelor interatomice.

MAGE poate arăta diferite conformații ale unei molecule.

Această „capacitate de animație moleculară" a programului MAGE nu este valabilă şi pentru programul Rasmol.

În ultimii ani au fost create programe de vizualizare grafice relativ necostisitoare, accesibile populației interesată de cercetarea moleculară cum ar fi RASMOL, Kinemage şi Chemscape Chime Viewer accesibil la adresa http://www.mdli.com./chemscape/chime.

Grafica moleculară este pasul spre dimensiunea a treia: analiza 3D, manipularea în 3D, precum şi compararea diferitelor conformații.

Animația joacă un rol esențial în vizualizarea simulărilor de dinamică moleculară.

Echipamentul hardware, parte componentă a tehnologiei de grafică moleculară, este completat de software-ul adecvat, necesar unor tehnici curente de vizualizare: zooming (alterarea scalei de vizualizare), clipping (simularea mişcării în planul Z al monitorului, care permite vizualizarea în adâncime), precum şi algoritmi de rotație-translație, acuratețea geometrică de reprezentare a atomilor şi lungimilor de legătură, manipularea precisă a obiectelor în procesul de superpoziție atomică, vizualizarea spațiului torsional (diagrama Ramachandran).

Alte suprafețe şi volume vizualizate curent sunt orbitalele moleculare, potențialul electrofilic sau lipofilic proiectate pe suprafața van der Waals sau pe suprafața accesibilă solventului (de tip "Connolly"). Suprafața poate fi la rândul ei opacă, translucentă sau reprezentată sub formă de puncte. Suprafețele Connolly sunt adesea vizualizate în combinație cu potențialul electrostatic.

Pentru proteine există convenții de vizualizare a structurii secundare folosind obiecte grafice de tip „panglică" sau „cilindru".

Programele de modelare moleculară sunt capabile să execute asemenea operații fie pe baza structurii predefinite (în format PDB), fie pe baza unor reguli care se referă la unghiurile de torsiune phi-psi, putând fi astfel vizualizată cu claritate structura secundară, terțiară sau cuaternară a unei proteine.

3.5. Exemplu de explorare a structurilor proteice cu PDB (Protein Data Bank)

- Accesați adresa de internet http://www.rcsb.org/pdb/home/home.do
- căutați "Protein Kinase C Interacting Protein with PDB-ID 1AV5" (figura 3.5.a).

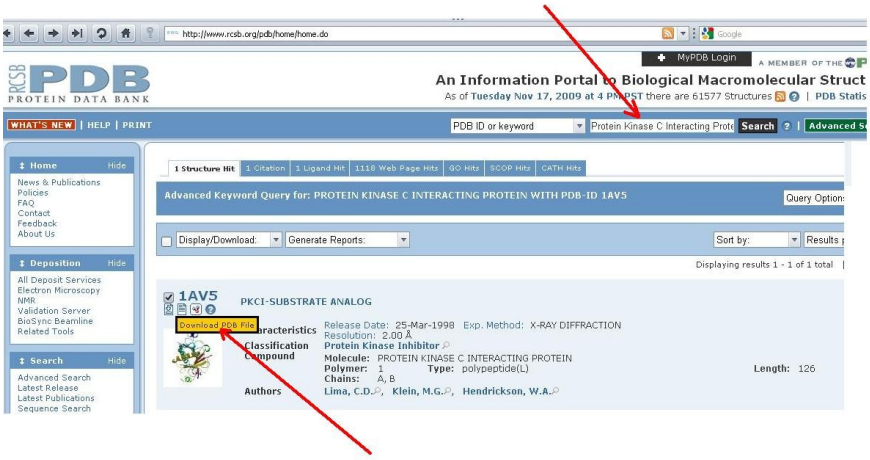


Figura 3.5.a. Interogare cu PDB

- Salvați structura în format.pdb
- Structura terțiară vizualizată cu jmol o alegeți ca în figura 3.5.a, iar rezultatul e prezentat în figura 3.5.b.

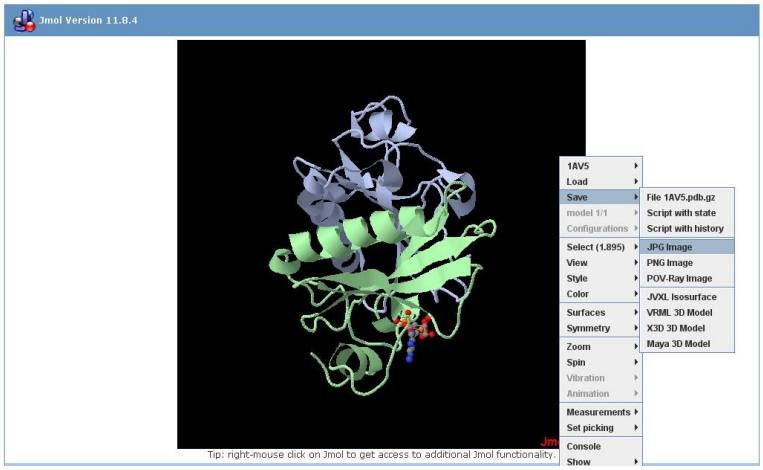


Figura 3.5.b. Exemplu de structură terțiară vizualizată cu jmol

3.6. Utilizarea Cn3D

Acest modul de program îl puteți accesa de pe adresa http://www.ncbi.nlm.nih.gov/sites/gquery (figura 3.6.a).

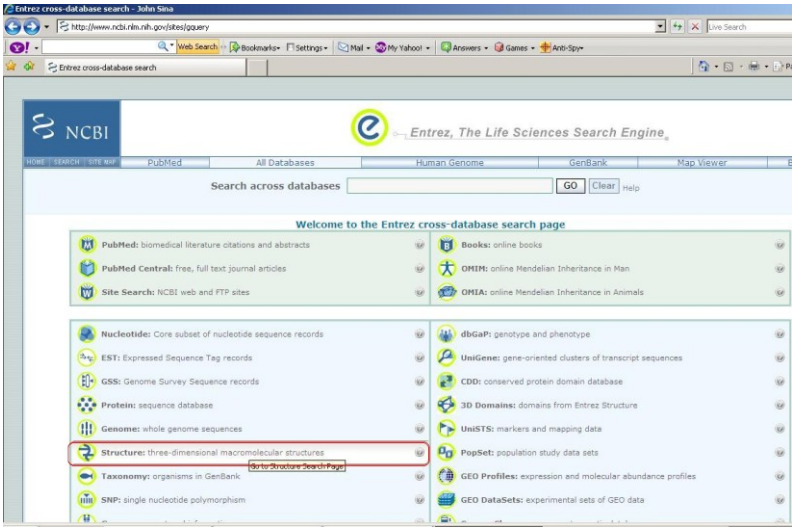


Figura 3.6.a. Opțiunea de vizualizare a structurii 3D moleculare

În căsuța "search" vom scrie *fibrinogen* şi vom obține rezultatul căutării în figura 3.6.b:

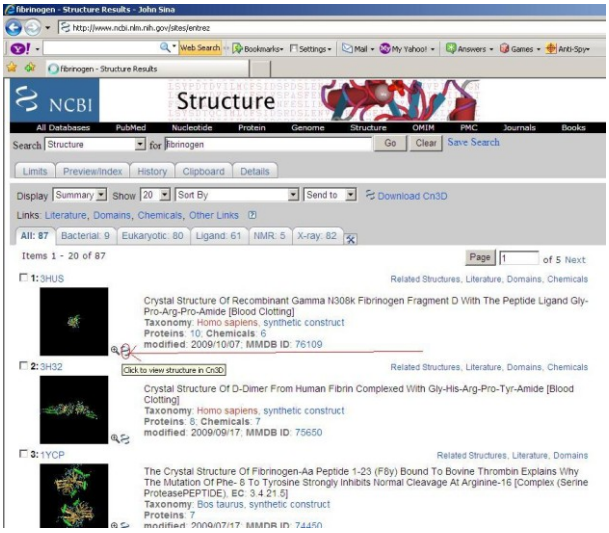


Figura 3.6.b. Rezultatele interogării

Prin accesarea butonului indicat de săgeată (figura 3.6.b) se va deschide programul Cn3D (figura 3.6.c).

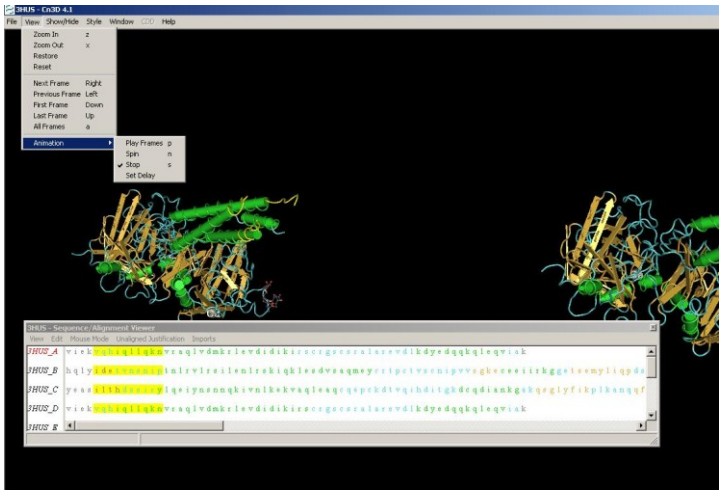


Figura 3.6.c. Vizualizarea celor 3 structuri ale fibrinogenului

În Cn3D vom avea afişată structura secundară şi terţiară în ecranul principal, precum şi structura primară (lanţul de aminoacizi) în ecranul "Sequence alignment viewer" (figura 3.6.c). În acest ecran aminoacizii sunt organizaţi pe domenii care corespund vederii 3D a proteinei din background. Tot în Cn3D avem la dispoziţie meniul "View", din care putem alege *zoom in* (pentru a vedea anumite domenii ascunse), precum şi opţiunea *animantion* prin care putem crea o animaţie în care structura 3D a proteinei se va roti. De asemenea avem la dispoziţie meniul "Styles" unde putem selecta opţiunea *Rendering shortcuts*. Aceasta va afecta modul de vizualizare al proteinei, fiind disponibile următoarele variante (figura 3.6.d): **worms** – vizualizarea cu spirale si panglici (cea mai sugestivă); **tubes; wire; ball and stick** - vizualizează atomii şi legăturile dintre molecule.

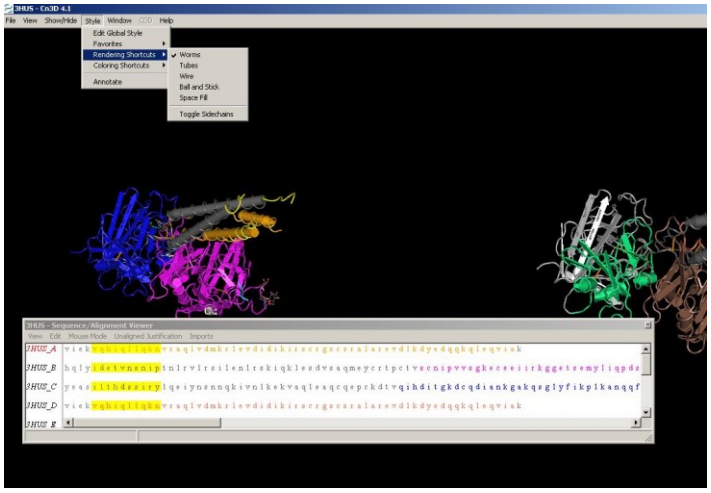


Figura 3.6.d. Moduri diferite de vizualizare ale structurilor terţiare şi cuaternare

Tot din meniul „Styles" avem opţiunea *Coloring Shortcuts* pentru a selecta culoarea de reprezentare a structurii 3D:

- **Secondary structure** - va colora structura secundară
- **Domain** - va colora un domeniu (partea funcţională a proteinei)

- **Molecule** - va colora o anumită moleculă
- **Temperature** - va colora în funcție de temperatură
- **Charge** - va colora în funcție de încărcătura electrică
- **Hidrophobycity** - va colora în funcție de hidrofobicitate.

3.7. Exemplu de utilizare a programului RasMol

Denumirea **RasMol** provine de la ***ras**ter display of **mol**ecules*. **Raster** este un tip de afişaj computerizat folosit în special pentru a afişa suprafețe solide. RasMol deschide fişiere de tipul.**pdb** (**p**rotein **d**atabase **b**ank).

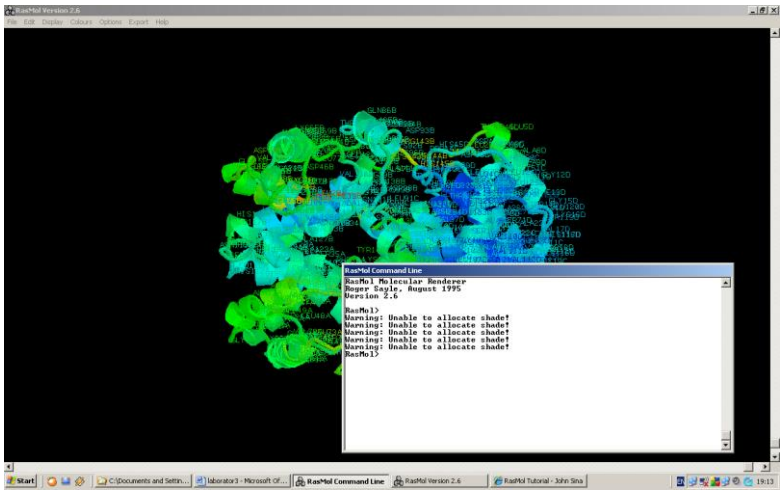


Figura 3.7.a. Vizualizarea proteinei cu programul RasMol

Fereastra "Linia de Comanda" apare în fața ecranului principal (figura 3.6.a). În continuare ne propunem să introducem câteva comenzi în liniile de comandă:

RasMol > ***help***
Citiți introducerea din fereastra de *help.*
RasMol > ***help commands***

Windows: Main Window
Ecranul principal este ecranul pe care va fi afişată structura proteinei. Se recomandă ca ecranul cu linia de comandă să fie aşezat în josul ecranului principal. Cu acest aranjament făcut se pot vedea şi ultimele linii de comandă şi mesajele trimise de RasMol. În ecranul principal se prezintă o structură sub formă de cadru de sârmă a unui model de proteină (figura 3.7.a).

Display: Backbone
Aşa se pot vedea doar atomii de carbon din poziția alpha. Liniile care conectează atomii de *Carbon alpha* sunt virtuale de obicei. Se pot vizualiza elemente de structură secundară ca elice aplha sau ca foițe beta pliate.
Încercați şi alte comenzi din meniul "Display". Fiecare comandă din acest meniu modifică felul în care RasMol reprezintă sau afişează modelul. La final afişați modelul *backbone*. Astfel structura secundară va fi şi mai evidentă.

Colours: Structure
Această comandă colorează elicea alpha în roz, foițele beta în galben şi alte părți de moleculă în gri deschis.
RasMol > ***select hetero and not hoh***
Această comandă va selecta numai grupările hetero (non-proteice) din acest fişier, excluzând moleculele de apă care sunt frecvent incluse în modelele cristal. Nimic nu se întâmplă până nu este introdusă o altă comandă. **Atenție! Comanda afectează numai atomii selectați în mod curent.** De asemenea nu este nevoie de a aduce fereastra cu linia de comandă în primul plan pentru a introduce comenzi.

Display: Ball & Stick
RasMol afişează un model de "bețe" şi "mingi" ale atomilor selectați. Observați că această comandă nu afectează lanțul proteic pentru că nu e selectat.

Colours: CPK
RasMol colorează atomii selectați conform unor convenții chimice larg răspândite: Carbonul (C) este gri deschis, Azotul (N) se colorează în albastru deschis, iar oxigenul se colorează în roşu.
Dați click pe Carbonul din poziția alpha a proteinei şi urmăriți linia de comandă. Această manevră se numeşte „extragerea unui atom". Când veți alege un atom din fereastra principală RasMol îl va identifica (figura 3.5.b). O să observați o expresie de genul: *Atom: CA 652 Group: PRO 81* care ne spune că am selectat carbonul din poziția alpha (CA) din restul 81 în citocromul b5, care este o prolină. RasMol defineşte restul " ca un "grup" (de aminoacizi), iar porțiunea non-proteică dintr-o moleculă ca "hetero". Dacă vedeți un atom colorat în galben după modelul "ball and stick" dați click pe el. Va fi afişat ceva de genul: *Atom: FE 754 Hetero: HEM 201*. Atomul galben numit FE este Fierul (Fe3+, de fapt) în centrul grupului hem al citocromului b5. Folosind comanda "select hetero and not hoh" şi afişând totul într-un stil contrast (similar cu displaylul Display:Ball&Stick) putem găsi foarte rapid grupările nonproteice din fişierul nostru PDB. Selectând un atom dintr-un grup putem afla numele în PDB al acelui grup, fapt care ne va folosi mai târziu.

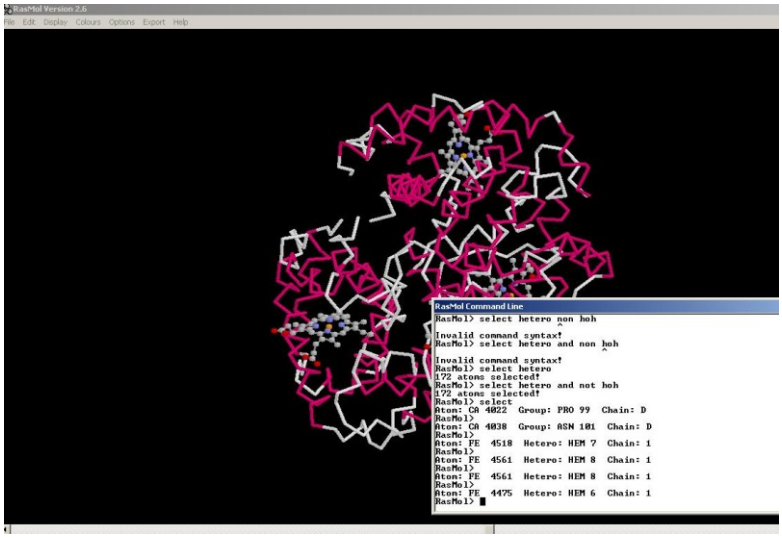


Figura 3.7.b. Tipuri de vizualizări ale atomilor şi fereastra de comandă

Atenţie! Selectarea unui atom funcţionează cel mai bine cu modul de vizualizare **Wireframe** şi **Sticks** şi nu funcţionează deloc cu modul **Ribbons**, **Strands** şi **Cartoons** (fiind inexact prin vizualizarea cu **Spacefill** şi **Ball&Stick**). Dedesubt este afişată lista de comenzi utilizată până acum (figura 3.5.b). Puteţi folosi această listă de comenzi pentru a va crea cât mai repede o idee asupra oricărei macromolecule pentru care aveţi fişierul PDB cât şi pentru a afişa eventualele grupări hetero (cofactori, inhibitori, apa):
Display: Backbone
Colours: Structure
Rasmol > **select hetero and not hoh**
Display: Ball & Stick
Colours: CPK.

3.8. Exerciţii propuse

1. Descărcaţi RasMol de la adresa http://www.umass.edu/microbio/rasmol/

2. Din ce clasă face parte proteina "Protein Kinase C Interacting Protein with PDB-ID 1AV5"?

R: Face parte din familia inhibitorilor de protein kinaza.

3. Căutaţi şi vizualizaţi hexokinaza umană I (Swissprot cod HXK1_HUMAN, număr P19367, PDB-id 1HKB).

4. Vizualizaţi domeniul de legare cu beta-D –glucoza cu ajutorul programului Ligand explorer.

R:Dupa ce aţi căutat proteina 1HKB la secţiunea Ligand Chemical Component=> beta-D- glucose accesaţi Ligand explorer.

5. Folosiţi GOR IV pentru a prezice structura secundară a proteinei cu structura primară următoare (lanţul alpha al hemoglobinei):

vlspadktnv kaawgkvgah ageygaeale rmflsfpttk tyfphfdlsh gsaqvkghgk kvadaltnav ahvddmpnal salsdlhahk lrvdpvnfkl lshcllvtla ahlpaeftpa vhasldkfla svstvltsky r

(http://npsa-pbil.ibcp.fr/cgi in/npsa_automat.pl?page=npsa_gor4.html)

6. Comparaţi rezultatele cu structura vizualizată în Cn3D a proteinei de mai sus (ID 2HCO).

Exerciţii cu RasMol

Mergeţi la link-ul PDB şi descărcaţi:

1ZNI (insulina porcina)

1LPH (insulina umana creata artificial)

Daţi dublu click pe icoana *Raswin* de pe desktop. Va apărea fereastra *RasMol* şi fereastra cu linia de comandă RasMol minimalizată. Accesaţi fereastra cu linia de comandă RasMol pentru a afişa linia de comandă. Scrieti ***background white***, apasaţi "enter". Se va schimba fundalul ecranului principal.

Duceţi fişierul PDB 1ZNI în fereastra principală RasMol. Acum puteţi vedea structura insulinei afişată cu displayul wireframe în ecranul principal.

Câți aminoacizi are poteina (insulina)?

Câte lanțuri polipeptidice are?

Care este aminoacidul din capătul N-terminal şi cel din capătul C-terminal al lanțurilor A şi B?

Care este structura secundară?

Care este cel mai lung Helix?

R: Câte lanțuri polipetidice are insulina?

Introduceți următoarele comenzi în linia de comandă:

reset
restrict backbone
ribbons on
wireframe off
color chain

Puteți salva un script pe hard-disk introducând comanda: ***write script C:\zni.sp***

R: Câți aminoacizi are lanțul A?

select *.ca and *a (selectează toți atomii CA din lantul A)

(Răspuns: "21 atoms selected!", deci există 21 de aminoacizi în lanțul A)

R. Care este capătul N-ternimal şi C-terminal?

select 1,21 and *.ca and *a label

R: Care este structura secundară? Care este cel mai lung Helix ?

Există pantru lanțuri polipeptidice în structura insulinei, fiecare afişat cu o culoare diferită. Un fişier PDB foloseşte o literă unică pentru fiecare lanț polipeptidic, începând cu litera "A".

De exemplu introduceți comanda: ***restrict *a***

Se va afişa numai lanțul „A" în ecranul principal RasMol.

R: Cum puteți folosi RasMol pentru a afla capătul N-terminal şi C-terminal al lanțului A?

Accesați ambele capete ale lanțului şi veți vedea afişat în linia de comadă RasMol, aminoacizii corespunzători.

4. Aplicația Vector NTI – programul Align X

4.1. Obiectivele lucrării de laborator

- compararea secvențelor ADN şi a secvențelor proteice
- descrierea generală a unei proteine, determinarea numărului de aminoacizi din care e formată, funcția ei, greutatea moleculară şi încărcarea electrică
- alinierea proteinelor
- vizualizarea arborelui filogenetic
- afişarea aliniamentului folosind matricele de scor.

Align X este un program, parte componentă din Vector NTI care poate fi folosit pentru a compara secvențe de ADN sau de proteine.

Pentru a instala programul Vector NTI suite 8 vom accesa următoarea adresă:

http://www.bioinformatics.uthscsa.edu/www/genomics/windows.html

De asemenea se pot descărca şi tutorialele. După ce lansăm în execuție programul VectorNTI, dăm click pe butonul *"Local Database"* (figura 4.1.a). Vom parcurge această lucrare prin intermediul a 3 exemple.

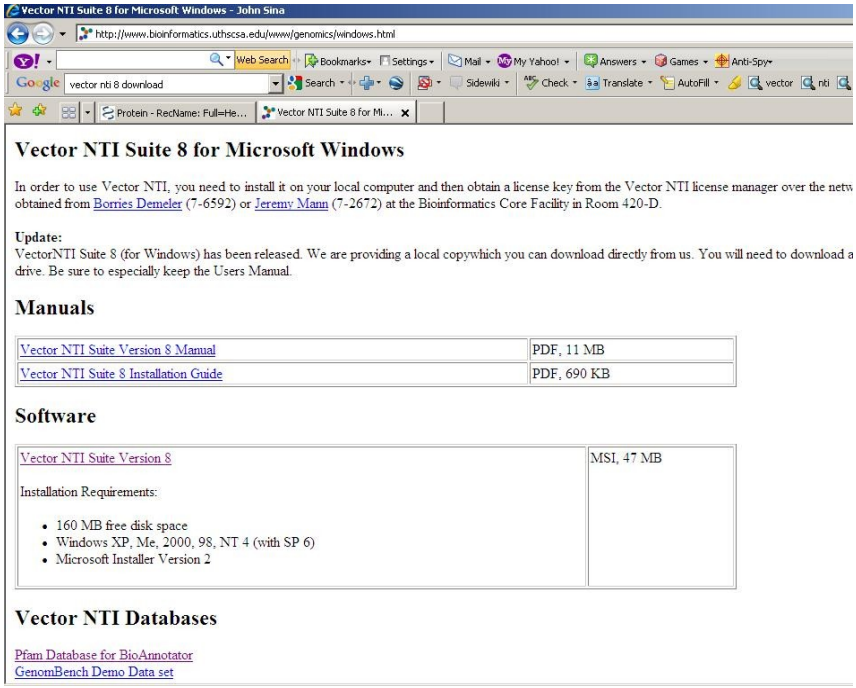


Figura 4.1.a. Modul de descărcare al programului Vector NTI

4.2. Exemplul 1 – Ne propunem să creăm o secvență de nucleotide pentru care să vizualizăm matricea DotMatrix

Dacă dorim să vizualizăm componentele bazei de date trebuie să alegem opțiunea "Exploring" din meniul principal al programului (figura 4.2.a).

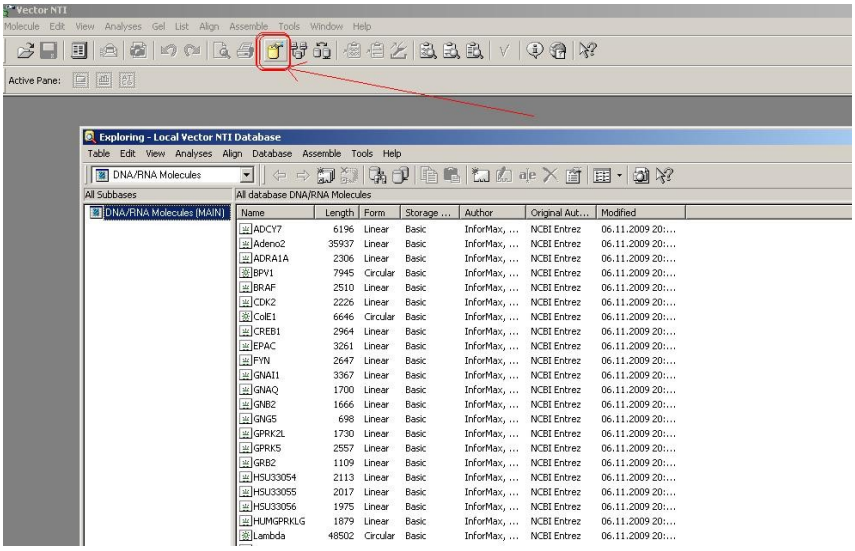


Figura 4.2.a. Afișarea fișierelor cu molecule ADN/ARN din BD a programului

Selectăm DNA/RNA molecules, facem click dreapta şi selectăm "New Item" pentru a crea o nouă moleculă, căreia îi dăm numele "test".

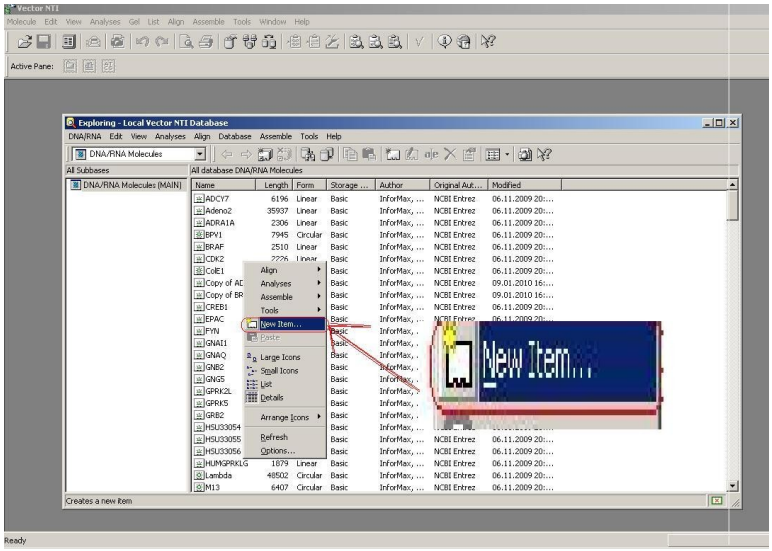


Figura 4.2.b. Exemplu de creare a unei noi molecule

Se va deschide o fereastră în care vom putea crea o moleculă test pentru care vom afişa alinierea vizuală a nucleotidelor din exemplul de la curs (Dot Matrix) - vom numi noua moleculă creată "**Test**" (figura 4.2.c).

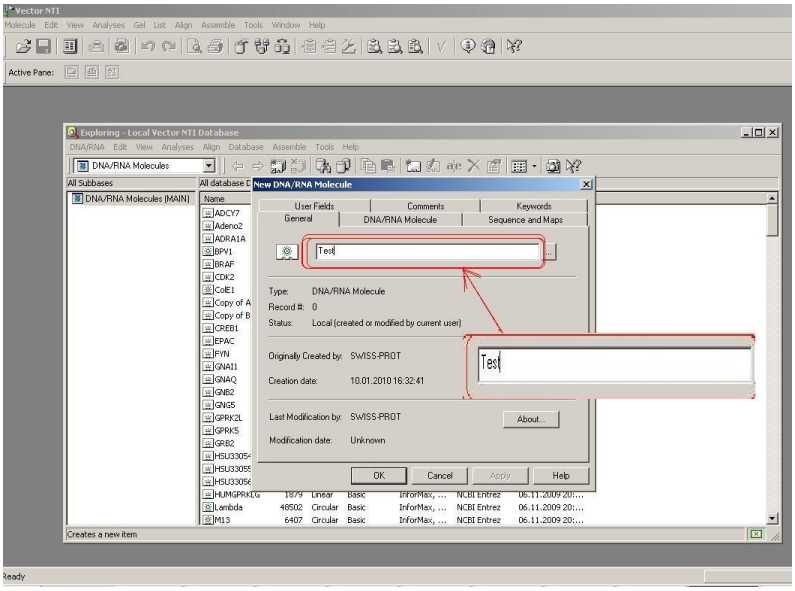


Figura 4.2.c. Denumirea noii molecule

Pentru a completa cu informaţie noua moleculă daţi click pe tab-ul ***Sequence and Maps***, iar după aceea click pe butonul ***Edit Sequence*** şi introduceţi următoarea secvenţă de nucleotide **ACCTGAGCTCACCTGAGTTA** (figura 4.2.d)**.**

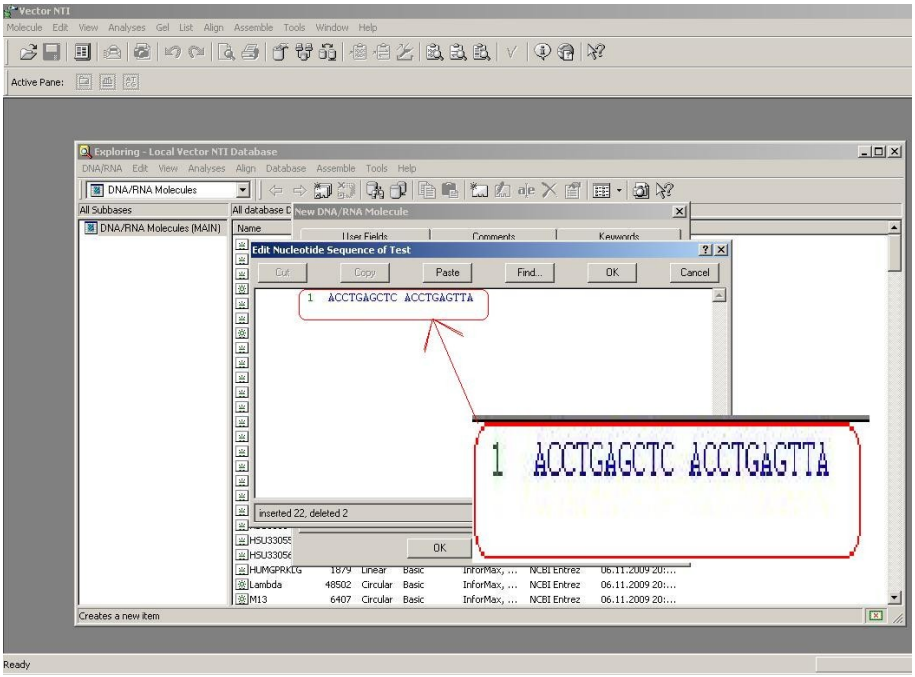


Figura 4.2.d. Introducerea secvenţei de nucleotide în fereastra de editare a programului

Apoi apăsați tasta OK, după care faceți click dreapta pe molecula **Test** apoi alegeți opțiunea **Align X** şi **Align Selected Molecules.**

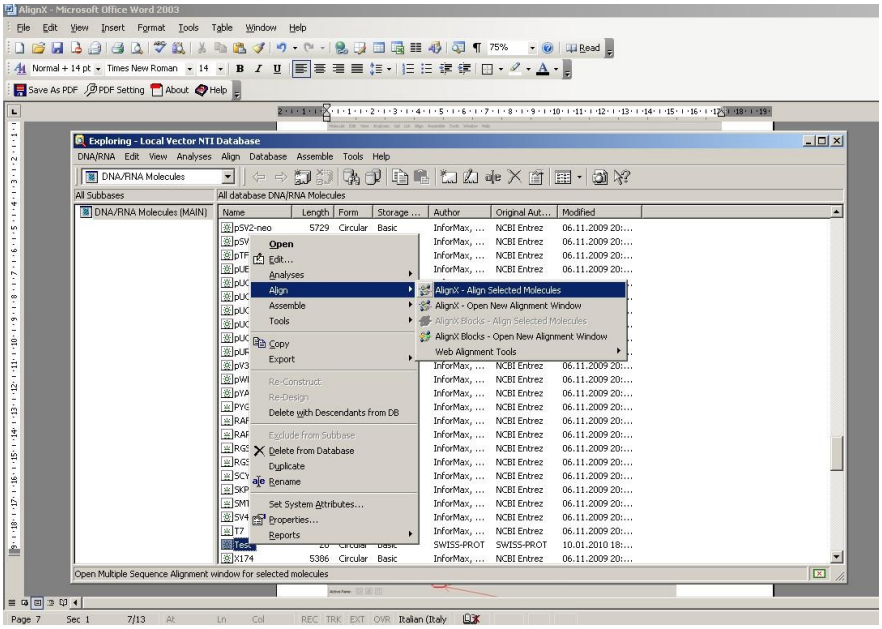


Figura 4.2.e. Opțiunile de aliniere a secvențelor

Se va deschide fereastra Align X iar pentru a afişa alinierea vizuala (Dot Matrix) pentru secvența "test" vom alege această opțiune (figura 4.2.e).

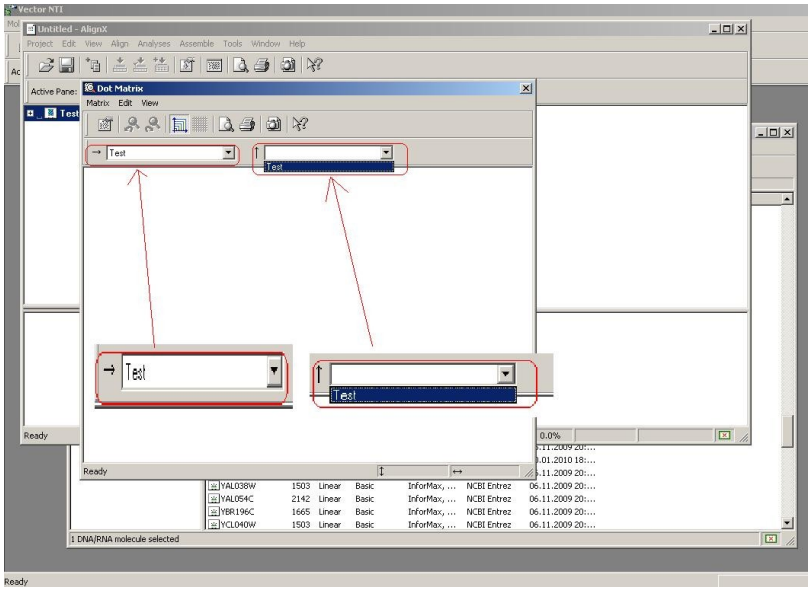


Figura 4.2.e. Alinierea secvenței "test" cu ea însăşi

Se va afişa fereastra cu Dot Matrix. Observăm că pe orizontală vor fi afişate nucleotidele moleculei **Test**. Vom alege astfel încât şi pe verticală să fie afişate nucleotidele moelculei **Test.**

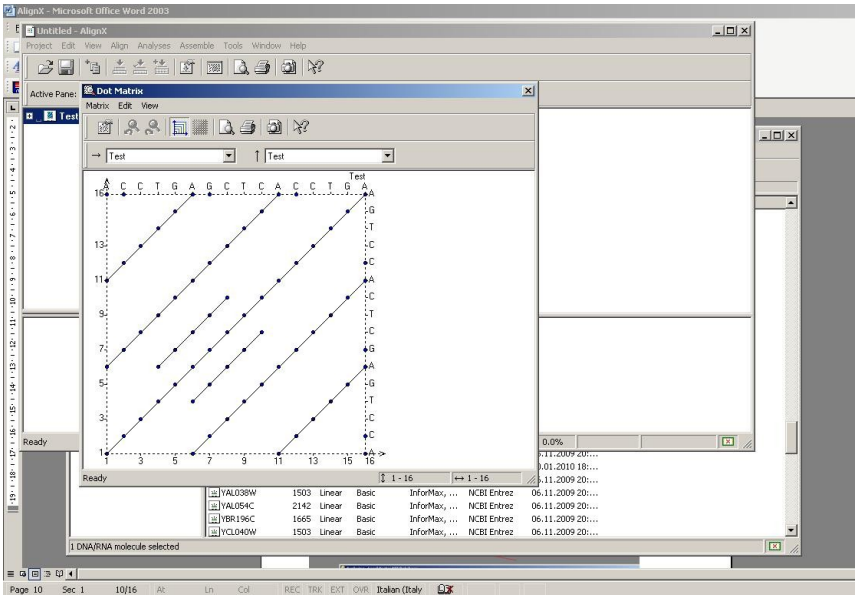


Figura 4.2.f. Conținutul matricei Dot Matrix

Observăm că atât pe orizontală cât şi pe verticală se află doar 16 nucleotide din cele 20 introduse. Pentru aceasta vom alege din meniul *Matrix =>Matrix setup => window* iar în caseta *window* vom introduce valoarea 1.

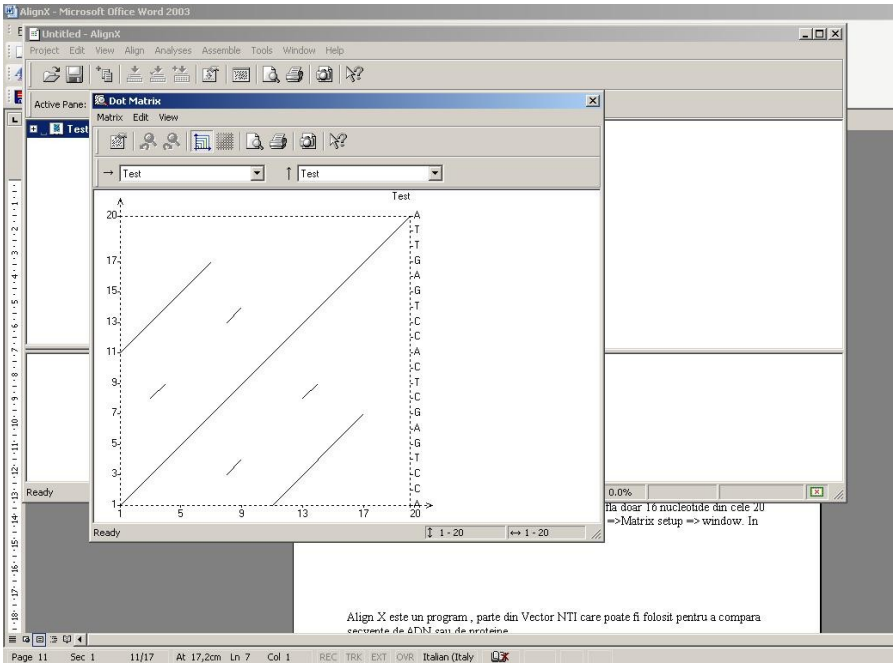


Figura 4.2.g. Reprezentare finală a Dot Matrix

4.3. Exemplu 2 – Compararea a două proteine cu programul Vector Align X

Din secțiunea *Database explorer* selectăm protein.

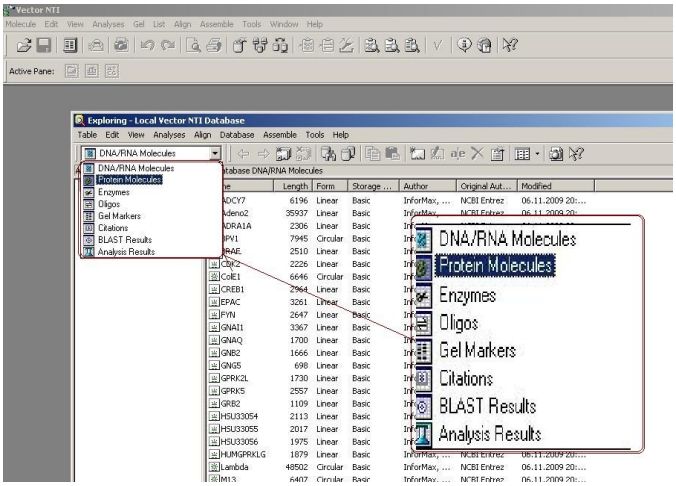


Figura 4.3.a. Selecția proteinelor

Apoi selectăm 2 molecule: **41BB_HUMAN** şi **4F2_HUMAN** (figura 4.3.b).

Pentru alinierea lor facem click dreapta şi apoi tastăm *Align => Align selected molecules*.

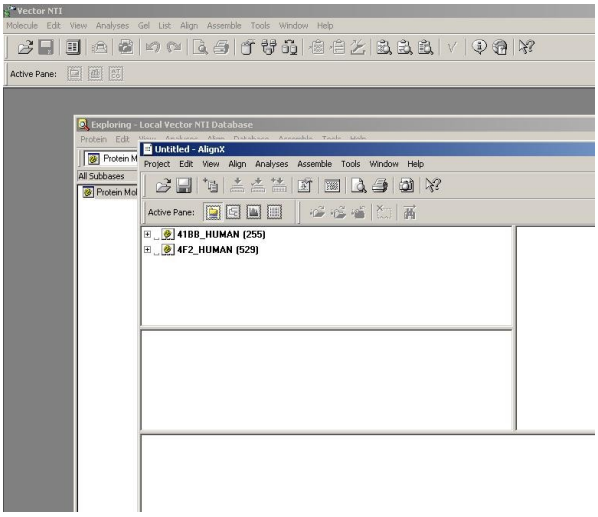


Figura 4.3.b. Cele două molecule alese spre comparare, în fereastra AlignX

Se va deschide subprogramul alignX. Daca vom da click pe „+" vom putea obține următoarele informații despre proteine: descrierea generală a proteinei, numărul de aminoacizi din care e formată, funcția, greutatea moleculară şi încărcarea electrică.

Vom selecta apoi DotMatrix, unde vom vedea secvențele de proteine comparate (figura 4.3.c).

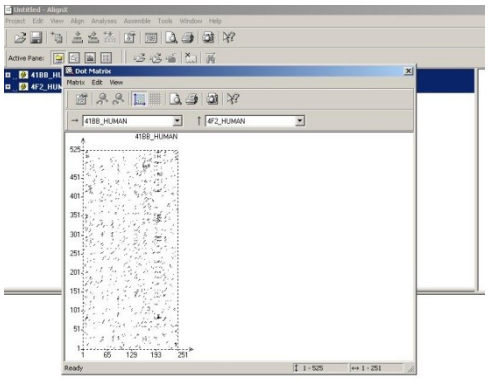


Figura 4.3.c. DotMatrix pentru alinierea celor două secvenţe

4.4. Exemplul 3

La fel ca în exemplul 2, după ce deschidem programul Vector NTI, din secţiunea *Database explorer* alegem opţiunea "protein" după care vom selecta următoarele proteine: **41BB_HUMAN, 5H1A_HUMAN, 5H1A_MOUSE** şi **5H1A_RAT.**

Executăm click dreapta => *Align X => Align selected molecules* pentru a alinia secvenţele proteinelor de tip 5H1A. Va apărea următorul ecran care conţine următoarele panouri ca în figura 4.4.a:

- Panoul cu text (Text Pane)
- Panoul care conţine arborele filogenetic (phylogenetic Pane)
- Panoul care conţine aliniamentul (alignament Pane)
- Panoul care conţine analiza (Analisys Pane) sau panoul grafic.

Selectăm moleculele şi apăsăm butonul *Align*.

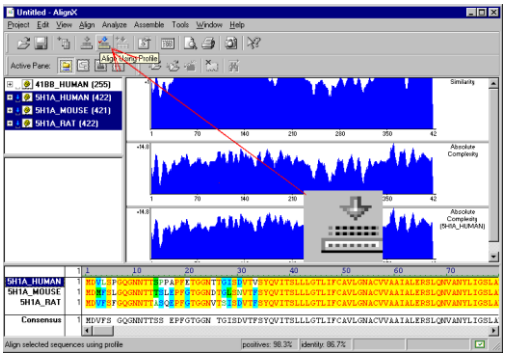


Figura 4.4.a. Panouri cu vizualizarea secvenţelor aliniate

În panoul *"Text"* daţi dublu click pe o moleculă şi vizualizaţi datele despre aceasta. Aceste informaţii conţin tipul moleculei, forma moleculei, comentarii, referinţe şi date în format GenBank.

Panoul implicit de analiză conţine trei reprezentări grafice ale rezultatelor aliniamentului.

- Primul grafic conţine profilul calităţii aliniamentului. Se analizează dacă secvenţa este identică, similară sau slab similară cu secvenţa de comparat.

- Al doilea grafic conține semnificația statistică a profilului (complexitatea absolută a unui aliniament). Este suma tuturor scorurilor calculate conform matricei de substituție.
- Al treilea grafic prezintă semnificația statistică a aliniamentului pentru o moleculă selectată în raport cu molecula de comparat (figura 4.4.a).

Putem adăuga analize adiționale prin tastarea butonului *"Graphics Pane"*. Apoi vom apăsa pe *"View" => list of analyses*.

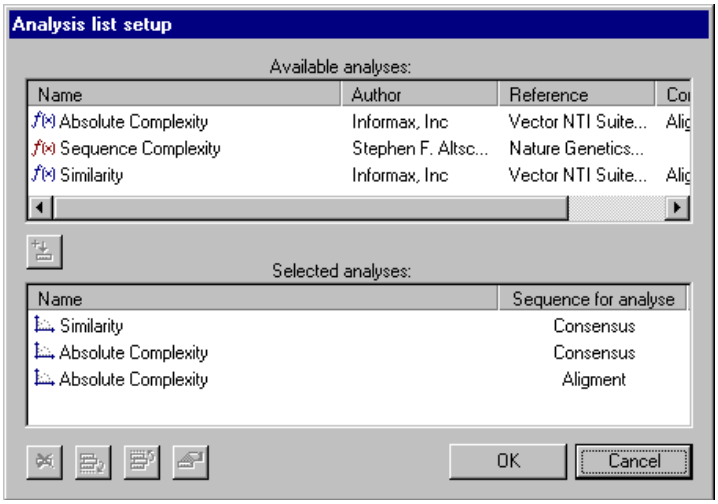


Figura 4.4.b. Lista analizelor adiționale

Sunt afişate toate analizele disponibile pentru ADN sau proteine, depinzând de tipul moleculelor de aliniat.

Analiza filogenetică este o metodă de studiere a presupuselor relații evolutive ale proteinelor. Panoul care conține arborele filogenetic îl va afişa doar dacă sunt comparate cel puțin 4 molecule (proteine) – figura 4.4.c.

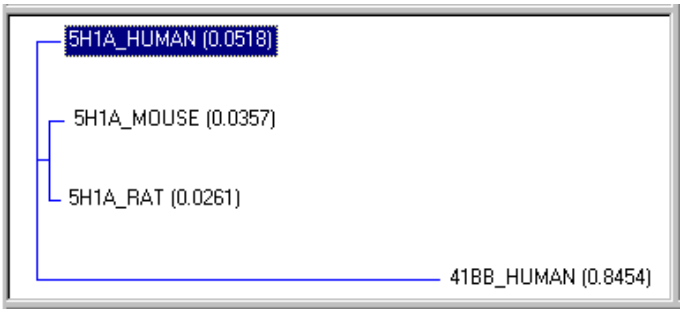


Figura 4.4.c. Arborele filogenetic pentru cele 4 molecule selectate inițial

Arborele filogenetic este creat folosind metoda ***Neighbor Joining*** (NJ) of Saitou and Nei.

Setarea parametrilor aliniamentului se obține prin intermediul butonului *"alignament setup"*. Aici vom putea seta parametrii pentru *Pairwise Alignment*, *Multiple sequence alignment*, precum şi *Score Matrix* (matricea de scor) – figura 4.4.d.

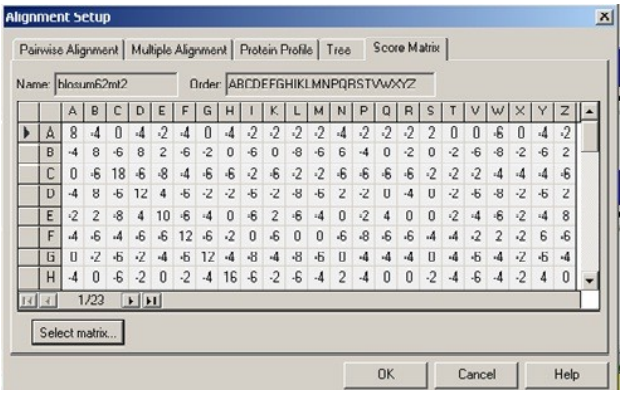


Figura 4.4.d. Afişarea elementelor matricei de scoruri BLOSUM62

Din tab-ul *score matrix* vom apăsa pe butonul *select matrix* şi vom selecta matricea de scor pe care vrem să o folosim. Matricea de scor se prezintă ca un fişier text (figura 4.4.d).

Putem selecta matricele de tipul BLOSUM, PAM, DAYHOFF sau cu ajutorul utilitarului *matrix editor*, putem crea propria noastră matrice de scor.

Editarea aliniamentului se realizează prin butonul *Edit alignment*. Aici vom putea selecta o secvență de aminoacizi şi o vom putea muta în cadrul aliniamentului (figura 4.4.e).

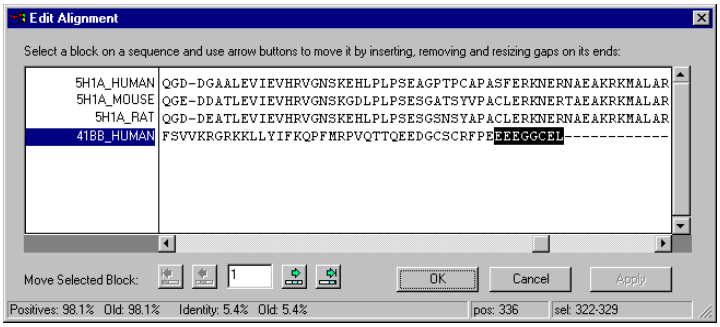


Figura 4.4.e. Exemplu de editare în cadrul aliniamentului secvenţelor selectate

4.5. Exerciții propuse

1. Să se selecteze cele 3 proteine de tipul 5H1D şi proteina 5H1E, să se alinieze aceste proteine, după care să se vizualizeze arborele filogenetic. Afişați aliniamentul folosind matricele de scor PAM 120, PAM 50 şi BLOSSUM 55.

2. Selectați proteinele de tipul _ECOLI (esicheria coli), aliniați-le şi vizualizați arborele filogenetic. Să se afişeze aliniamentul folosind matricele de scor PAM10, PAM20, PAM40, BLOSUM 100, BLOSUM 85.

3. Să se selecteze proteinele de tipul YLR şi să se alinieze aceste proteine. Vizualizați arborele filogenetic şi afişați aliniamentul folosind matricele de scor PAM 120, PAM 50 si BLOSSUM 55.

4. Afişați DOT MATRIX pentru proteinele din exercițiile de mai sus (2 câte 2).

5. Cu ajutorul algoritmului ***Needleman-Wunsch*** din fişierul excel calculaţi matricea de scoruri asociată alinierii celor două secvenţe de nucleotide: GAATTCAGTTA şi GGATCGA.

R:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Potriviri:	5			Key:		Red: move diagonally up and left, sequences are aligned										
2	Nepotriviri	-3					Green: move up, gap in top (query) sequence										
3	Gap (InDel)	-4					Blue: move left, gap in left-side (subject) sequence										
4							White: move up or left or diagonally, alternate optimal alignments										
5	**Algoritmul Needleman-Wunsch**																
6			G	A	A	T	T	C	A	G	T	T	A				
7		0	-4	-8	-12	-16	-20	-24	-28	-32	-36	-40	-44	-48	-52	-56	-60
8	G	-4	5R	1←	-3←	-7←	-11←	-15←	-19←	-23	-27←	-31←	-35←	-39←	-43←	-47←	-51←
9	G	-8	1	2R	-2	-6	-10	-14	-18	-14R	-18←	-22←	-26←	-30←	-34←	-38←	-42←
10	A	-12	-1↑	6R	7R	3←	-1←	-5←	-9	-13←	-17	-21	-17R	-21←	-25←	-29←	-33←
11	T	-16	-7↑	2↑	3	12R	8	4←	0←	-4←	-8	-12	-16←	-20	-24	-28	-32
12	C	-20	-11↑	-2↑	-1	8↑	9R	13R	9←	5←	1←	-3←	-7←	-11←	-15←	-19←	-23←
13	G	-24	-15	-6↑	-5	4↑	5	9↑	10R	14R	10←	6←	2←	-2←	-6←	-10←	-14←
14	A	-28	-19↑	-10	-1R	0↑	1	5↑	14R	10	11R	7	11R	7←	3←	-1←	-5←
15		-32	-23↑	-14↑	-5↑	-4	-3	1↑	10↑	11R	7	8R	7↑	16R	12	8	4
16		-36	-27↑	-18↑	-9↑	-8	-7	-3↑	6↑	7	8R	4	5R	12	21R	17	13
17		-40	-31↑	-22↑	-13↑	-12	-11	-7↑	2↑	3	4	5R	1	10R	17	26R	22
18		-44	-35↑	-26↑	-17↑	-16	-15	-11↑	-2↑	-1	0	1	2R	6	15R	22	31R
19		-48	-39↑	-30↑	-21↑	-20	-19	-15↑	-6↑	-5	-4	-3	-2	7R	11	20R	27
20		-52	-43↑	-34↑	-25↑	-24	-23	-19↑	-10↑	-9	-8	-7	-6	3	12R	16	25R
21		-56	-47↑	-38↑	-29↑	-28	-27	-23↑	-14↑	-13	-12	-11	-10	-1	8	17R	21
22		-60	-51↑	-42↑	-33↑	-32	-31	-27↑	-18↑	-17	-16	-15	-14	-5	4	13	22R

6. Aceeaşi aliniere realizaţi-o cu algoritmul ***Smith-Waterman.***

R:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Potriviri:	5			Key:		Red: move diagonally up and left, sequences are aligned										
2	Nepotriviri	-3					Green: move up, gap in top (query) sequence										
3	Gap (InDel)	-4					Blue: move left, gap in left-side (subject) sequence										
4							White: move up or left or diagonally, alternate optimal alignments										
5	**Algoritmul Smith-Waterman**																
6			G	A	A	T	T	C	A	G	T	T	A				
7		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	G	0	5R	1←	0	0R	0R	0R	0R	5R	1←	0	0R	0R	0R	0R	0R
9	G	0	5R	2R	0	0R	0R	0R	0R	5R	2R	0	0R	0R	0R	0R	0R
10	A	0	1↑	10R	7R	3←	0←	0R	5R	1	2R	0R	5R	1←	0	0R	0R
11	T	0	0	6↑	7R	12R	8	4←	1↑	2R	6R	7R	3←	2R	0	0R	0R
12	C	0	0R	2↑	3	8↑	9R	13R	9←	5←	2↑	3	4R	0	0R	0R	0R
13	G	0	5R	1←	0	4↑	5	9↑	10R	14R	10←	6←	2←	1R	0	0R	0R
14	A	0	1↑	10R	6	2←	1	5↑	14R	10	11R	7	11R	7←	3←	0←	0R
15		0	0	6↑	7R	3	0	1↑	10↑	11R	7	8R	7↑	16R	12	8	5R
16		0	0R	2↑	3	4R	0	0	6↑	7	8R	4	5R	12	21R	17	13
17		0	0R	0↑	0	0	1R	0	2↑	3	4	5R	1	10R	17	26R	22
18		0	0R	0R	0R	0R	0	0R	0↑	0	0	1	2R	6	15R	22	31R
19		0	0R	0R	0R	0R	0R	0R	0R	0R	0R	0	0	7R	11	20R	27
20		0	0R	0R	0R	0R	0R	0R	0R	0R	0R	0R	0R	5R	12R	16	25R
21		0	0R	0R	0R	0R	0R	0R	0R	0R	0R	0R	0R	5R	10R	17R	21
22		0	0R	0R	0R	0R	0R	0R	0R	0R	0R	0R	0R	5R	10R	15R	22R

5. Clustal X

5.1. Obiectivele lucrării de laborator

- alinierea multiplă a secvențelor de proteine cu programul **ClustalX**
- vizualizarea arborelui filogenetic cu acelaşi program
- setarea parametrilor aliniamentului
- scriere aliniamentului ca fişier Postscript.

5.2. Introducere

Programul CLUSTALX oferă o interfață grafică de utilizator pentru algoritmul CLUSTALW de aliniere multiplă a secvențelor (MSA – Multiple Sequences Alignament).

Publicațiile originale care descriu CLUSTALX şi CLUSTALW sunt unele dintre cele mai citate documente - atunci când au fost publicate au prevăzut o creştere majoră în acuratețea şi rapiditatea cu care utilizatorii ar putea construi MSAs, cu accent pe secvențele de proteine - în cazul CLUSTALX a fost adăugată în plus o interfață uşor de folosit (GUI) care să conducă la o mare popularitate a programelor.

Există în prezent o serie de alte produse software MSA care sunt, datorita multiplelor seturi de secvențe şi analize, o alegere mai bună decât CLUSTALX dacă țelul vostru este de a obține o aliniere (MSA) a unui set de secvențe de proteine - şi, desigur, dacă sunteți interesați de alinierea secvențelor ADN (ceva ce CLUSTALW/X nu au avut ca obiectiv principal). Cu toate acestea CLUSTALX rămâne util deoarece în unele circumstanțe face o bună aliniere a secvențelor (judecate după standardele de azi) şi poate cel mai important din cauza mai multor caracteristici ale GUI care oferă interfețe uşor de utilizat pentru mai multe sarcini cheie care sunt necesare, în multe cazuri, pentru a construi o bună aliniere multiplă a secvențelor (MSA).

5.3. Utilizarea Clustal X

ClustalX poate fi descărcat de la adresa: http://www.clustal.org/download/current/.
Observație! Dacă obțineți eroare de tipul "lipsă fişier mingw10.dll", la rularea programului ClustalX, o puteți corecta prin descărcarea acestui fişier de la adresa: http://www.dll-files.com/dllindex/dll-files.shtml?mingwm10 după care îl salvați în locația C:\Windows\System.

O vedere generală asupra programului ClustalX o oferă imaginea din figura 5.3.a.

Bara de meniu-oferă acces la o gamă largă de caracteristici şi acțiuni care pot fi aplicate la aliniere: de exemplu, salvarea unei alinieri, ajustarea sistemului de colorat, etc.

Căsuța de alegere a modului - este folosit pentru a comuta între mai multe tipuri de alinieri: alinierea multiplă şi alinierea după profil (aşa cum este arătat mai sus).

Reţineţi că unele opţiuni de meniu sunt disponibile numai în alinierea multiplă sau numai în alinierea după profil.

Figura 5.3.a. Interfaţa cu utilizatorul a programului ClustalX

Căsuţa cu mărimea fontului este folosită pentru a specifica dimensiunea fontului utilizat pentru a reprezenta secvenţele în **zona de afişare a aliniamentului**.

Numele de identificare al secvenţelor folosite în aliniament sunt afişate în **Panoul cu numele secvenţelor**.
Linia de consens oferă un rezumat al gradului de conservare a reziduurilor din coloana de aliniere corespunzătoare:

- „*” (reziduuri identice în toate secvenţele)
- „:” (conservate bine în coloană)
- „.” (conservate slab în coloană).

Liniarul de aliniere indică poziţia în aliniere (prima coloană are atribuită poziţia 1, etc.) a unei coloane date.

Zona de conservare a alinierii oferă o diagramă bară care indică gradul de conservare al fiecărei coloane în aliniament.

5.4. Crearea fişierului de intrare pentru alinierea multiplă

Acest program, ca oricare program de calculator are nevoie ca datele pe care le utilizează (fişier de intrare) să fie într-un format pe care să îl recunoască. Pentru a creea fişierul de intrare puteţi folosi un editor de texte, de preferabil Notepad. Clustal citeşte fişiere într-unul din cele 7 formate de fişiere de intrare, înlocuind orice secvenţă care este deja încărcată. Toate secvenţele trebuie să se găsească într-un singur fişier, secvenţă după secvenţă. Formatele de fişier care sunt recunoscute automat sunt:

NBRF/PIR, EMBL/SWISSPROT, Pearson (Fasta), Clustal (*.aln), GCG/MSF (Pileup), GCG9/RSF şi GDE. Toate caracterele non-alfabetice (spaţii, cifre, semne de punctuaţie) sunt ignorate, exceptând „-“care e folosită pentru a indica un GAP („.” în MSF/RSF).

Programul încearcă sa recunoască automat diferitele tipuri de fişiere folosite şi să stabilească dacă secvenţele sunt de aminoacizi sau de nucleotide.

FASTA şi NBRF/PIR sunt recunoscute pentru că au caracterul „>" ca primul caracter din fişier.

EMBL/Swiss Prot sunt recunoscute după literele "ID" de la începutul fişierului.

CLUSTAL este recunoscut după cuvântul "CLUSTAL" de la începutul fişierului.

GCG/MSF este recunoscut după unul din următoarele caractere:
- cuvântul "PileUp" la începutul fişierului
- cuvântul "!!AA_MULTIPLE_ALIGNMENT" sau "!!NA_MULTIPLE_ALIGNMENT" la începutul fişierului.
- cuvântul "MSF" la începutul primei linii din fişier şi caracterele ".." la sfârşitul acestei linii.
- cuvântul "!!RICH SEQUENCE" la începutul fişierului.

Dacă 85% sau mai mult dintre caracterele secvenţei sunt din A,C,G,T,U sau N atunci secvenţa va fi tratată ca o secvenţă de nucleotide.

5.4.1. Exemplu

Copiaţi secvenţa următoare în fişierul text:

gi|15146064|gb|AY040893.1| Homo sapiens individual VP37 mitochondrial control region

GGTCTATCACCCTATTAACCACTCACGGGAGCTCTCCATGCATTTGGTATTT
TCGTCTGGGGGGTGTGCA

CGCGATAGCATTGCGAGACGCTGGAGCCGGAGCACCCTATGTCGCAGTATC
TGTCTTTGATTCCTGCCTC

ATCCTGTTATTTATCGCACCTACGTTCAATATTACAGGCGAACATACCTACT
AAAGTGTGTTAATTAATT

AATGCTTGTAGGACATAATAATAACAATTG

gi|15146065|gb|AY040894.1| Homo sapiens individual VP5mitochondrial control region

GGTCTATCACCCTATTAACCACTCACGGGAGCTCTCCATGCATTTGGTATTT
TCGTCTGGGGGGTATGCA

CGCGATAGCATTGCGAGACGCTGGAGCCGGAGCACCCTATGTCGCAGTATC
TGTCTTTGATTCCTGCCTC

ATCCTATTATTTATCGCACCTACGTTCAATATTACAGGCGAACATACTTACT
AAAGTGTGTTAATTAATT

ClustalX poate recunoaşte mai multe formate de secvenţe, dar noi vom utiliza în exemplul următor formatul FASTA. Formatul FASTA poate fi recunoscut uşor deoarece prima linie începe cu caracterul „>". Această linie conţine titlul secvenţei. Secvenţa va începe de la linia următoare. Caracterul „>" va fi urmat de un singur cuvânt, pe care ClustalX îl va folosi ca nume al secvenţei în alinierea multiplă pe care o creează.

Numele fişierului este bine ca să aibă salvat ID-ul de tip Genbank pentru a putea fi uşor de recunoscut ulterior.

5.4.2. Alinierea multiplă

Programarea dinamică poate fi folosită pentru a alinia mai multe secvenţe. Poate fi creat un aliniament optim, dar nu pot fi folosite mai mult de 5 secvenţe, datorită timpului de calculare. De aceea este aplicată metoda progresivă a alinierii multiple.

Clustal efectuează o aliniere multiplă-globală după metoda progresivă. Etapele sunt următoarele:

a) Efectuează alinierea a câte două secvenţe folosind programarea dinamică,
b) Foloseşte scorul alinierii pentru a produce un arbore filogenetic cu metoda neighbour-joining (NJ),
c) Aliniază mai multe secvenţe folosind arborele filogenetic.

Deşi cele mai apropiate secvenţe sunt aliniate primele, iar apoi se adaugă secvenţe sau grupuri de secvenţe, utilizând aliniamentul iniţial pentru a produce o aliniere multiplă, arătând în fiecare coloană variaţia secvenţei în cadrul aliniamentului.

Cu cât sunt adăugate mai multe secvenţe la profil, gap-urile se acumulează şi influenţează alinierea secvenţelor următoare. Clustal calculează gap într-un mod nou, proiectat să plaseze gap-urile între domeniile conservate. Gapurile găsite în aliniamentul initial rămân fixate. Pot fi adăugate gap-uri noi, când sunt adăugate secvenţe noi, dar gap-urile nu pot fi şterse, numai adăugate. Clustal de asemenea implementează metode care încearcă să compenseze matricea scor (de ex. PAM), gap-urile care pot fi prevăzute şi diferenţele în lungimea secvenţei.

Clustal are opţiuni avansate:

- Adaugă secvenţe cu greutate
- Adaugă greutăţi la poziţii diferite în secvenţă
- Adaugă o secvenţă sau o aliniere la alinierea existent
- Foloseşte arborele definit de utilizator pentru aliniere.

Unele dintre acestea vor fi discutate în capitolele următoare.

Problema cu alinierea progresivă este reprezentată de dependenţa ultimei secvenţe din aliniamentul multiplu de aliniamentele iniţiale. Primele secvenţe care vor fi aliniate sunt cele mai strâns legate din arborele secvenţial. Dacă aceste secvenţe se vor alinia bine, vor exista câteva erori în aliniamentele iniţiale. Cu toate acestea, cu cât sunt mai departe înrudite secvenţele, cu atât mai multe erori vor fi făcute, şi aceste erori vor fi propagate la aliniamentul secvenţial multiplu.

O a doua problemă a metodei alinierii progresive este alegerea potrivită a matricelor scor, a penalizării gap-urilor care se aplică la setul de secvenţe.

5.4.3. Introducerea datelor în programul ClustalX

După ce porniți ClustalX, ecranul va arăta în felul următor (figura 5.4.3.a):

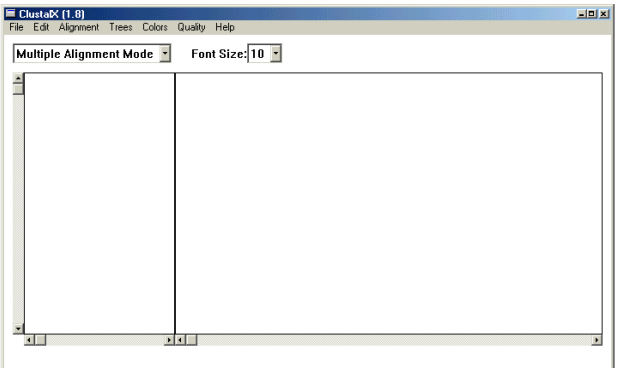


Figura 5.4.3.a. Fereastra programului Clustal

Alegeți din meniul File opțiunea *Load Sequences*. Pe urmă alegeți folderul care conține fișierul de intrare (fișierul text care conține secvențele pentru aliniere în diferite formate FASTA).

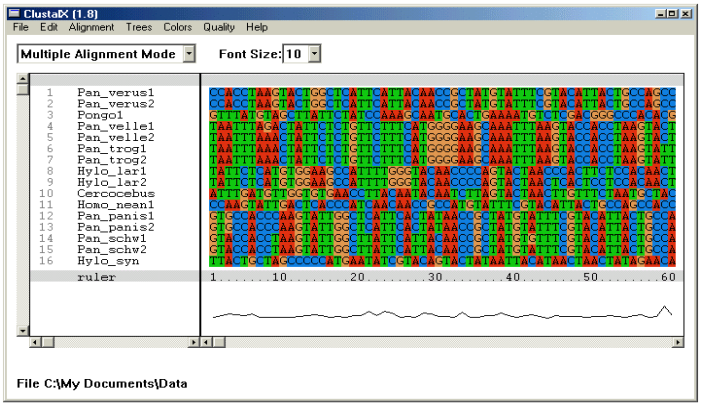


Figura 5.4.3.b.Exemplu de secvențe de nucleotide sub formă de fișier text

Panoul din stânga (figura 5.4.3.b) afișează secvențele conform cu numele care urmează după caracterul „>” din fișierul de intrare. Panoul din dreapta afișează începutul fiecărei secvențe. Puteți derula mai în dreapta pentru a vedea rezultatul fiecărei secvențe folosind bara de scroll din josul panoului.

5.5. Setarea parametrilor aliniamentului

Aliniamentul este realizat după câteva etape succesive:

- Resetarea tuturor Gap-urilor (*Alignment->Alignment parameters*, *Edit->Remove all Gaps*)
- Rafinarea parametrilor aliniamentului pereche (*Alignment->Alignment parameters*)
- Rafinarea parametrilor aliniamentului multiplu (*Alignment->Alignment parameters*)
- Rafinarea formatului fișierului de ieșire (*Alignment->Output Format Options*)

- Scrierea aliniamentului ca un fişier Postscript (*File->Write Alignment as Postscript*)
- Evaluarea calităţii aliniamentului:
  a. Dacă nu sunteţi mulţumiţi -> mergeţi la pasul 1
  b. Dacă sunteţi mulţumiţi -> rafinaţi aliniamentul manual.

5.5.1. Parametrii alinierii pereche

Pentru a crea alinierea pereche, ClustalX trebuie să ştie ce penalităţi să aplice pentru crearea fiecărui gap şi pentru extensia acelui gap. Alegeţi *Pairwise Alignment Parameters* din meniul *Alignment*. Veţi vedea apoi o căsuţă dialog asemănătoare cu figura următoare.

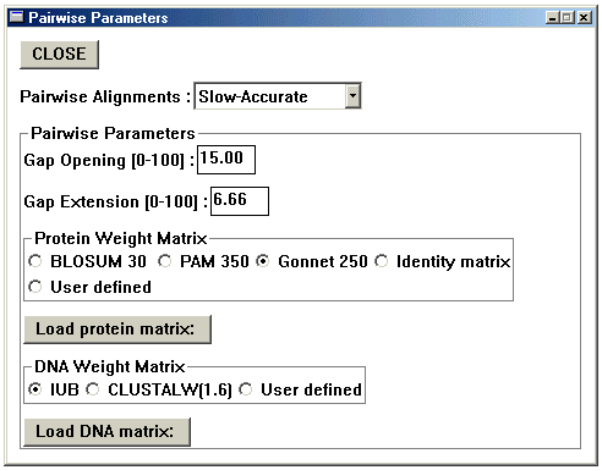


Figura 5.5.1.a. Afişarea parametrilor alinierii pereche

Prima opţiune în cadrul alinierii pereche vă dă posibilitatea să alegeţi dintre metodele **Slow-accurate** (mai lentă, dar precisă) şi **Fast-approximate** (rapidă, dar aproximativă). Metoda Slow este preferată, dar dacă aliniaţi mai multe secvenţe, sau secvenţele de aliniat sunt foarte lungi, programul va rula într-un timp mai lung, aşadar veţi dori să alegeţi metoda Fast. Metoda Fast foloseşte o metodă K-tuple pentru alinierea pereche, spre deosebire de metoda Slow care foloseşte algoritmi din programarea dinamică. Căsuţa afişează valorile implicite pentru penalităţile Gap Opening si Gap Extension (deschiderea şi întinderea gap-urilor). Micşorând penalităţile gap veţi putea să introduceţi mai multe gap-uri şi mai puţine nepotriviri. Acest lucru va avea ca rezultat potriviri care nu reflectă omologia. Creşterea penalităţilor pentru gap va avea un efect invers, dar s-ar putea să omitem potriviri care sunt omoloage.

Parametrii matricelor scor pot fi modificaţi şi ei. Parametrii matricei IUB DNA are ca scor pentru potrivire 1.9 şi pentru nepotrivire 0. Matricele de scor pentru proteine sunt echivalente aceloraşi matrice folosite ca modele evoluţionare în producţia dendrogramelor. Toate matricele scor au avantajele şi dezavantajele lor: PAM a fost folosită mult timp, dar este oarecum depăşită, iar Gonnet poate fi mai potrivită să o folosiţi. Matricea BLOSUM pare a fi cea mai bună pentru căutarea în cadrul bazelor de date. Puteţi crea şi încărca propria dumneavoastră matrice în ClustalX. Pentru descrierea formatului matricei puteţi să studiaţi fişierul *matrices.h* care arată în felul următor:

- Pentru aminoacizi:

```
char *amino_acid_order = "ABCDEFGHIKLMNPQRSTVWXYZ";
short blosum30mt[]={
4,
0, 5,
-3, -2, 17,
0, 5, -3, 9,
0, 0, 1, 1, 6,
-2, -3, -3, -5, -4, 10,
0, 0, -4, -1, -2, -3, 8,
-2, -2, -5, -2, 0, -3, -3, 14,
0, -2, -2, -4, -3, 0, -1, -2, 6,
0, 0, -3, 0, 2, -1, -1, -2, -2, 4,
-1, -1, 0, -1, -1, 2, -2, -1, 2, -2, 4,
1, -2, -2, -3, -1, -2, -2, 2, 1, 2, 2, 6,
0, 4, -1, 1, -1, -1, 0, -1, 0, 0, -2, 0, 8,
-1, -2, -3, -1, 1, -4, -1, 1, -3, 1, -3, -4, -3, 11,
1, -1, -2, -1, 2, -3, -2, 0, -2, 0, -2, -1, -1, 0, 8,
-1, -2, -2, -1, -1, -1, -2, -1, -3, 1, -2, 0, -2, -1, 3, 8,
1, 0, -2, 0, 0, -1, 0, -1, -1, 0, -2, -2, 0, -1, -1, -1, 4,
1, 0, -2, -1, -2, -2, -2, -2, 0, -1, 0, 0, 1, 0, 0, -3, 2, 5,
1, -2, -2, -2, -3, 1, -3, -3, 4, -2, 1, 0, -2, -4, -3, -1, -1, 1, 5,
-5, -5, -2, -4, -1, 1, 1, -5, -3, -2, -2, -3, -7, -3, -1, 0, -3, -5, -3, 20,
0, -1, -2, -1, -1, -1, -1, -1, 0, 0, 0, 0, 0, -1, 0, -1, 0, 0, 0, -2, -1,
-4, -3, -6, -1, -2, 3, -3, 0, -1, -1, 3, -1, -4, -2, -1, 0, -2, -1, 1, 5, -1, 9,
0, 0, 0, 0, 5, -4, -2, 0, -3, 1, -1, -1, -1, 0, 4, 0, -1, -1, -3, -1, 0,2,4};
```

- Pentru ADN:

```
char *nucleic_acid_order = "ABCDGHKMNRSTUVWXY";
short clustalvdnamt[]={
10,
0, 0,
0, 0, 10,
0, 0, 0, 0,
0, 0, 0, 0, 10,
0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 10,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 10,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
```

5.5.2. Parametrii alinierii multiple

Alegeți opțiunea *Multiple Alignment Parameters* din meniul *Alignment* (figura 5.5.2.a).

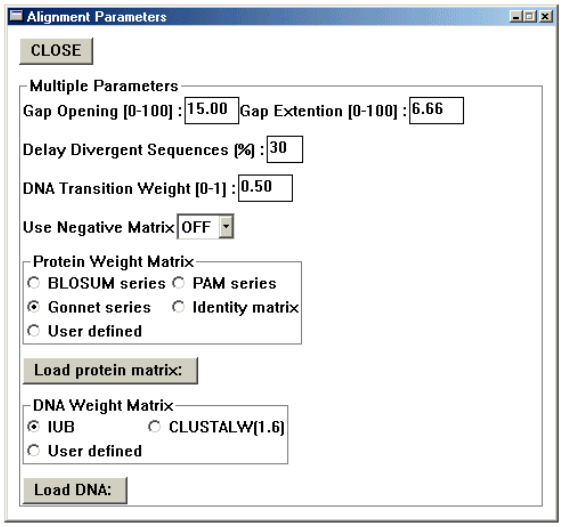


Figura 5.5.2.a. Fereastra de dialog pentru alinierea multiplă

Parametrii alinierii pereche şi multiple sunt configurați independent, deoarece Clustal are nevoie de amândoi. Cum a fost precizat anterior, o matrice a alinierii pereche este calculată inițial, iar apoi pe baza acelor distanțe este formată alinierea multiplă. Folosind diferite setări pentru acești paşi ai alinierii vom avea mai multă flexibilitate asupra modului în care este efectuată alinierea.

În comparație cu alinierea pereche vom avea câțiva parametrii în plus. *Delay Divergent Sequences* determină cum trebuie să fie două secvențe pentru ca aliniamentul lor să fie decalat. Această opțiune încearcă să compenseze pentru bias introdusă în metoda alinierii progresive. Scorul tranziției ADN poate fi modificat. Weight 0 înseamnă că tranzițiile sunt punctate ca şi nepotriviri, iar weight 1 înseamnă că tranzițiile au acelaşi scor ca transversiile. Pentru secvențe de ADN slab înrudite, scorul ar trebui să fie aproape de 0; pentru secvențe apropiate poate fi util să atribuim un scor mai ridicat. Nu mai trebuie sa alegeți individual matricea scor, pentru ca ştim deja cât de similare sunt secvențele. Clustal X va alege automat cea mai potrivită matrice scor dintr-o serie de matrice. Astfel, dacă ați schimbat matricea scor în cadrul parametrilor alinierii pereche, să efectuați şi aici aceeaşi schimbare.

5.5.3. Formatul de ieşire al alinierii

Ultimul lucru care trebuie modificat înaintea efectuării aliniamentului este formatul de ieşire.

Formatul fişierului de ieşire se poate modifica selectând *Output Format Options* din meniul *Alignment*.

Când ClustalX creează o aliniere, scrie secvențele aliniate într-un fişier. Există mai multe tipuri de formate de ieşire, care vor fi necesare în funcție de programul care vreți să îl folosiți pentru analiza ulterioară.

Pentru construcţia arborelui filogenetic alegeţi *Phylip* ca format al fişierului de ieşire. Să nu uitaţi să scrieţi secvenţele aliniate şi într-un fişier de tip Clustal.

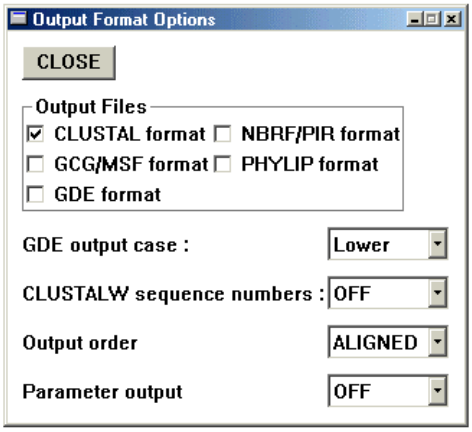


Figura 5.5.3.a. Opţiunile formatului de ieşire al alinierii

5.6. Crearea alinierii

Alegeţi opţiunea *Do Complete Alignment* din meniul *Alignment*. ClustalX va spune ce face în fiecare moment, dar nu ar trebui să folosiţi alt program în timp ce clustal efectuează alinierea. După ce alinierea a fost efectuată , ecranul principal va fi împrospătat cu secvenţele aliniate.

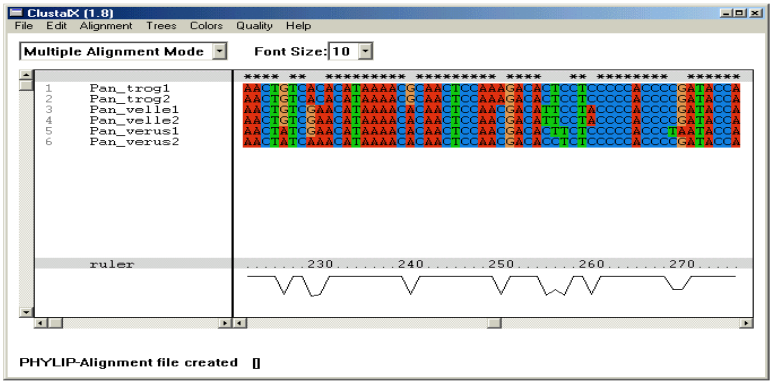


Figura 5.6.a. Exemplu de aliniere de tip Phylip

Bazele sunt colorate, ceea ce face ca evaluarea alinierii să fie mult mai uşoara. Histograma de sub liniar indică gradul de similaritate (figura 5.6.a). Vârfurile indică poziţiile cu cea mai mare similaritate, iar văile indică poziţiile cu cea mai mică similaritate. Linia gri aflată deasupra secvenţelor este folosită pentru a marca porţiunile conservate. Caracterul „*” indică poziţii care au fost conservate în totalitate (secvenţe identice).

5.7. Scrierea aliniamentului ca fişier Postscript

Este posibil să folosim fişierele pe care deja le-am creat pentru a construi un arbore filogenetic, dar calitatea şi valoarea acelui arbore nu va fi mai bună decât valoarea aliniamentului. Trebuie reținut că oricât de diferite ar fi două secvențe ClustalX va produce întotdeauna o aliniere. Doar faptul că există o aliniere nu înseamnă că secvențele sunt înrudite. Este la atitudinea utilizatorului să determine dacă secvențele din setul de date sunt omoloage, adică pot fi aliniate. Calitatea alinierii este mai uşor de verificat, dacă există copii tipărite ale secvențelor. Există două posibilități:

- puteți tipări alinierea în format ClustalX şi folosi aceasta, dar veți pierde informațiile colorate;
- puteți să instalați **Ghostscript** şi **Ghostview**, care vă vor permite să manipulați documentele Postcript şi să tipăriți alinierea ca postscript incluzând culorile.

Mergeți la meniul *File* şi selectați opțiunea *Write Alignment as Postscript* - figura 5.7.a.

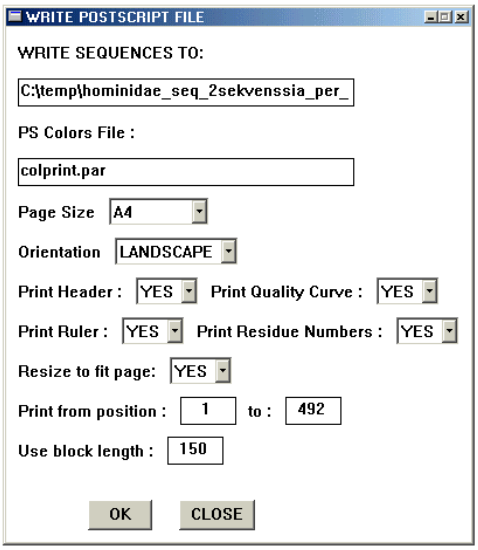


Figura 5.7.a. Exemplu de scriere a aliniamentului ca fişier Postscript

Ghostscript poate fi descărcat de la adresa:

http://sourceforge.net/projects/ghostscript/files/GPL%20Ghostscript/9.00/gs900w32.exe/download

şi **ghostview** de la adresa:

http://www.seas.ucla.edu/~ee5cta/ghostView/.

Vor trebui schimbate anumite opțiuni, iar după ce le-ați setat, apăsați "OK"; apoi deschideți fişierul postscript în Ghostview şi tipăriți alinierea folosind meniul *File=> Print.*

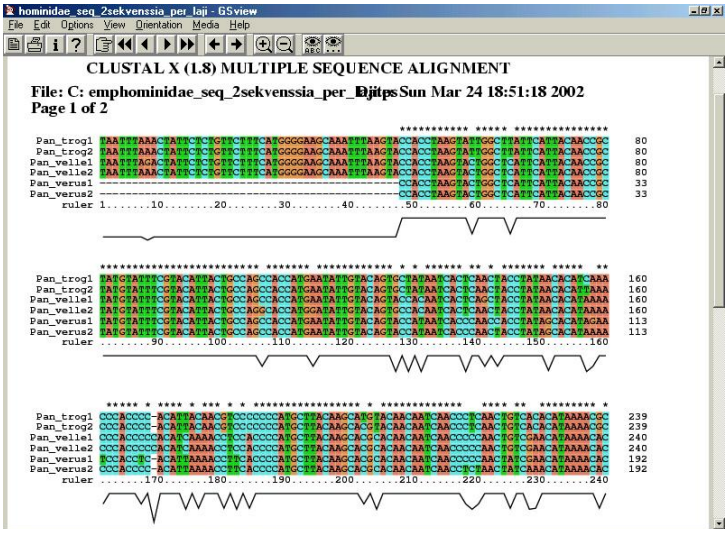


Figura 5.7.b Vizualizarea fişierului Postscript ce conţine alinierile secvenţelor, cu programul Ghostview

5.8. Arbori filogenetici folosind ClustalX

Pentru o analiză avansată a arborelui filogenetic se poate folosi pachetul **Phylip**, de la adresa:

http://bioweb.pasteur.fr/seqanal/phylogeny/phylip-uk.html.

NJ plot- este programul folosit de obicei cu ClustalX, de la adresa:

http://pbil.univ-lyon1.fr/software/njplot.html

Treeview este o altă metodă software de vizualizare, găsită la adresa:

http://taxonomy.zoology.gla.ac.uk/rod/treeview.html

ClustalX include implementarea algoritmului Neighbour-Joining (NJ) care ne permite să construim arbori filogenetici pornind de la o aliniere multiplă.

Atenţie: există o diferenţă între arborele ghid (care a fost construit pe baza distanţelor între perechi înaintea efectuării aliniamentului) şi arborele NJ (construit după aliniament). Arborele NJ este construit calculând distanţele dintre fiecare pereche de secvenţe în cadrul alinierii multiple. Alinierea dintre o pereche de secvenţe poate fi diferită în cadrul aliniamentului multiplu.

5.9. Exerciţii propuse

1. Vom porni de la un aliniament de secvenţe peptidice al proteinei *Homoserine O-succinyltransferases* care poate fi descărcată de la următorul link:

http://www.bigre.ulb.ac.be/Users/jvanheld/bioinformatics_introductory_course/web_course/data/sequence_analysis/metA_family.aln

2. Deschideţi acest fişier cu ClustalX. Observaţi că nu este nevoie să aliniaţi această secvenţă deoarece sunt deja aliniate.

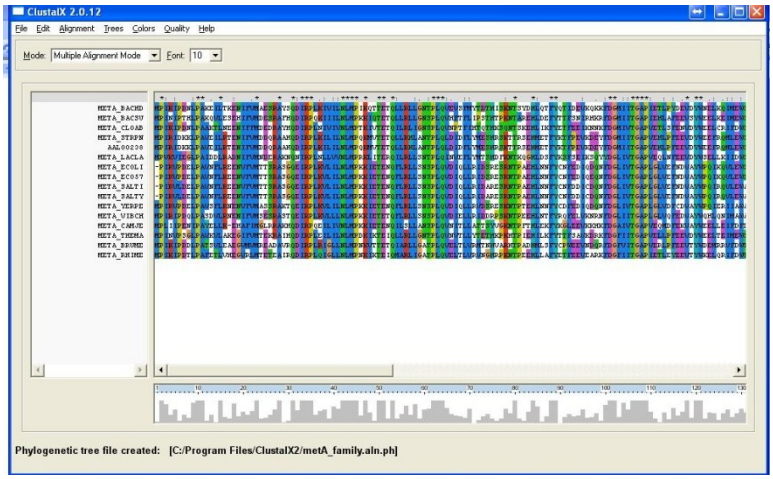


Figura 5.9.a. Aliniamentul de secvențe peptidice al proteinei *Homoserine O-succinyltransferases*

3. În meniul *Trees* selectați comanda *Draw Tree*. Aceasta va crea un fişier cu extensia .ph în directorul Clustal.

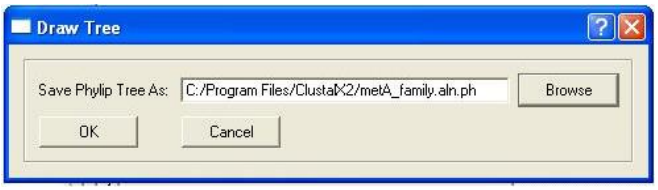
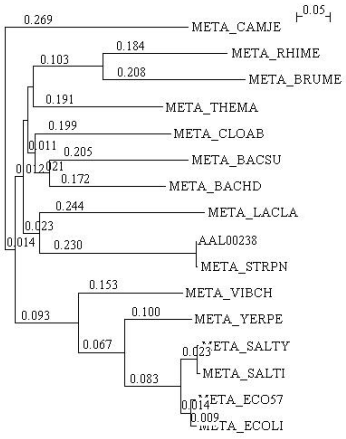


Figura 5.9.b. Exemplu de creare a fişierului cu descrierea arborelui filogenetic

4. Vom aplica şi procedura *bootstrap* pentru a estima gradul de înrudire dintre diferite ramuri ale arborelui. În meniul *Trees* apăsați *Bootstrap N-J tree*. Aceasta va crea un fişier cu extensia .php în acelaşi director cu aliniamentul Clustal.

5. Deschideți programul **Njplot**.

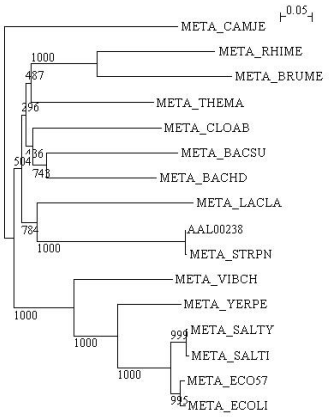
6. Folosind Njplot deschideți fişierul care conține arborele NJ (extensia .ph). Verificați opțiunea *Branch Lengths*.



R:

7. Folosind Njplot deschideți fișierul care conține arborele NJ bootstrapped. Verificați opțiunea *Bootstrap Values*.

R:



8. Deschideți diferitele tipuri de fișiere care conțin arborii filogenetic folosind sau Njplot sau Treeview.

- .dnd ClustalX guide tree
- .ph ClustalX neighbor-joining tree
- .phb ClustalX neighbor-joining bootstrap tree
- .tree the result of Phylip

Bibliografie

1. Lesk Arthur M. Introduction to Bioinformatics (3rd edition), Oxford Univ Press, Oxford UK, 2008
2. Durbin R., Eddy S., Krogh A., Mitchison G., Biological Sequence Analysis, Cambridge Univ Press, Cambridge Uk, 1998
3. Claverie J.M., Notredam C., Bioinformatics for dummies, Wiley, Hoboken N.J., 2007
4. Mihalaş G.I., Lungeanu Diana, Informatică medicală şi biostatistică, Ed. Victor Babeş, Timişoara, 2009
5. Brown T. A. Genomes, John Wilez & Sous, New York USA, 1999
6. Mihalaş G.I., Neagu M., Neagu A., Textbook of Biophysics, Ed. Eurobit, Timişoara, 2001
7. Swanson T. A., Kim S.I., Glucksman M.J., Biochemistry Molecular Biology and Genetics (4th edition), Wolters Kluwer, Philadelphia, 2007
8. Simon-Gruita A., Saitan T., Biologie, CD Press, Bucureşti, 2009
9. www.clustal.org –Jarno Tuimala-ClustalX
10. http://www.embl.de/~seqanal/courses/commonCourseContent/usingClustalx.html
11. http://www.bigre.ulb.ac.be/Users/jvanheld/bioinformatics_introductory_course/web_course/practicals/phylogeny.html#contents Jacques van Helden-bioinformatics_introductory_course