

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

Scuola di Scienze
Dipartimento di Fisica e Astronomia
Corso di Laurea in Fisica

Modelli simulativi di nodi con applicazione alla struttura 3D delle proteine

Relatore:
Prof. Daniel Remondini

Presentata da:
Adriano Zaghi

Correlatore:
Prof. Francesco Giacomini

Anno Accademico 2018/2019

Indice

| | | |
|----------|---|-----------|
| 1 | Introduzione | 2 |
| 1.1 | Teoria dei nodi e applicazioni | 2 |
| 1.2 | Definizione di nodo su varietà con estremi | 3 |
| 1.2.1 | Procedura | 6 |
| 1.3 | Proteine e nodi | 7 |
| 2 | Simulazione | 11 |
| 2.1 | Funzionamento della simulazione | 11 |
| 2.1.1 | Limiti | 15 |
| 2.2 | Risultati dello stiramento di toy-model | 16 |
| 2.2.1 | Rapporto tra numero di punti e tempo di calcolo | 16 |
| 2.2.2 | Limite di non compenetrazione | 18 |
| 3 | Risultati | 23 |
| 4 | Conclusioni | 33 |
| 4.1 | Possibilità di sviluppo | 34 |
| 5 | Ringraziamenti | 36 |

Capitolo 1

Introduzione

Nel presente studio si è sviluppato un programma in grado di simulare la dinamica di un filamento aperto, con due estremi. Si sono testate le potenzialità e i limiti di questa simulazione attraverso l'impiego di toy-model. Particolare attenzione ha richiesto il fenomeno della compenetrazione, ovvero il fatto che due segmenti del filamento simulato fossero in grado, in certe condizioni, di passare l'uno attraverso l'altro. Si è cercato il più possibile di evitare questo fenomeno. Ci si è concentrati sul fenomeno di annodamento a cui il filamento simulato può andare incontro, un fenomeno quotidiano ma non banale. Per comprenderlo si è partiti dalla definizione matematica di nodo, riferita a varietà chiuse, e la si è ricondotta a varietà con estremi. Si è dunque stati in grado di utilizzare il programma per ricercare la presenza di nodi nei sistemi simulati. Si è voluto testare la simulazione anche su dati reali e si è dunque studiato un database di proteine. Verrà spiegato nel capitolo 1.3 sotto quali aspetti la simulazione risulta efficace nel rappresentare questi oggetti. Nel database le proteine erano divise in base alle modalità di folding. Si è dunque valutata la correlazione tra la presenza di nodi e le modalità di questo processo, tipico delle proteine.

1.1 Teoria dei nodi e applicazioni

Si ha a che fare quotidianamente con quelli che vengono comunemente definiti nodi. Queste entità sono però state oggetto di una trattazione matematica approfondita solo dall'inizio del 900 ad oggi. In quest'ambito il nodo è definito come caratteristica topologica non locale di una varietà N dimensionale in uno spazio $(N+2)$ dimensionale, anche se la maggior parte degli studi sono incentrati su varietà unidimensionali in spazi tridimensionali. Nonostante nello studio dei nodi non siano implicati metodi matematici troppo avanzati, sono aperti ancora molti interrogativi nei loro riguardi. Parte della complessità del problema risiede certamente nel fatto che i nodi, pur avendo evidentemente degli aspetti che li accomunano, non possono comunque essere considerati tutti uguali.

Per questa ragione l'interesse matematico per la questione è stato in maggioranza rivolto a identificare un criterio univoco per distinguere i nodi, ordinarli in categorie e stimare quanti siano i nodi tra loro differenti che rientrino nelle categorie. È chiaro che non esiste limite alla complessità di un nodo e lo scopo finale di questo approccio matematico è quello di formulare una legge in grado di riempire tutte le categorie pensabili solo in base alla loro definizione e non allo studio di ogni singolo nodo. Dal punto di vista tecnico questi studi possono essere di diversa utilità a seconda del campo di applicazione, ma in ogni caso non possono che mostrare alcuni limiti. Primo fra tutti il fatto che tutte le definizioni e le teorie formulate sono applicabili esclusivamente a curve di Jordan, ovvero curve chiuse non autointersecanti, o a curve infinite non autointersecanti. Nulla o quasi nulla è detto su varietà che presentano estremi. Come vedremo infatti la presenza di estremi pone seri problemi a una definizione rigorosa di nodo. Inoltre affrontare il problema da un punto di vista esclusivamente topologico trascura completamente le caratteristiche fisiche e strutturali del supporto dei nodi come il diametro, la rigidità e la scabrosità. Queste caratteristiche, pur essendo irrilevanti nella classificazione di tutti i nodi immaginabili, rivestono un ruolo importante non appena si osservi esempi di nodi in natura o si cerchino loro applicazioni nella tecnica. Nella ricerca scientifica, uno degli ambiti in cui il tema dei nodi può rivestire una seria importanza è quello della biologia e in particolare lo studio delle proteine. La maggior parte di queste sono infatti costituite da lunghe catene di aminoacidi che possono, in linea di principio, essere soggette ad anodamento. Questa circostanza potrebbe influire sulle proprietà delle proteine. Al fine di fornire uno strumento di analisi efficace per questa ed altre branche della ricerca, l'obiettivo di questo studio è quello di sviluppare una simulazione virtuale di un filamento il più possibile realistica. Grazie ad essa si cercherà di rilevare la presenza di un nodo in una varietà data. Le varietà in esame sono dotate di estremi al fine di simulare una condizione più realistica e che abbia risvolti applicativi. Questo ha reso però necessaria una ridefinizione del concetto di nodo come entità topologica. Si è cercato di rimanere il più attinenti possibile alla definizione matematica al fine di poter sfruttare, sotto certe condizioni, definizioni e teoremi elaborati su varietà chiuse. Si è inoltre dotata la simulazione di parametri meccanici, ovvero forze di vario genere che agiscono sui punti materiali della corda con intensità regolabile, determinando l'evoluzione del sistema come entità fisica.

1.2 Definizione di nodo su varietà con estremi

Nonostante possa sembrare banale, è fondamentale stabilire con esattezza cosa, in questo studio, si intenda con nodo. L'oggetto della simulazione come detto non può supportare la definizione matematica di nodo poiché si tratta di una varietà non riconducibile né a una curva di Jordan né a una curva non limitata. Possiamo però confrontare la situazione

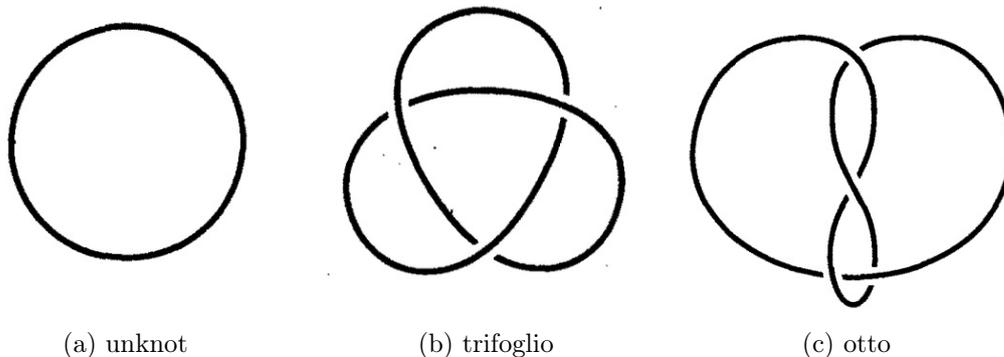


Figura 1.1: Sono riportati tre tipi di nodi su corde chiuse, tra cui quello triviale: l'unknot. È evidente come non sia possibile da uno ottenere l'altro attraverso trasformazioni di isotropia ambientale

simulata, che ha il pregio di essere più realistica, con la definizione topologica per tentare di allentare la rigidità di quest'ultima.

Ciò che consente in matematica di formulare una classificazione dei nodi è un principio di equivalenza, basato su una classe particolare di omeomorfismi: le deformazioni di isotopia ambientale. Questo particolare tipo di trasformazioni corrisponde sostanzialmente a quelle che sono compatibili con la continuità della posizione dei punti nello spazio e nel tempo, come ogni fenomeno di fisica classica. Preservano inoltre la continuità della varietà non permettendo la compenetrazione. Se la nostra varietà è illimitata o una curva di Jordan basta poco per rendersi conto che, data una configurazione di partenza, non potremo arrivare a qualsiasi altra usando solo questo tipo di trasformazioni. Ce ne si rende conto facilmente osservando *Figura 1.1*, la quale contiene tre nodi di tre tipi differenti.

Sono dunque identificabili insiemi di configurazioni isolati gli uni dagli altri, all'interno dei quali ci si può spostare da un elemento a un altro attraverso trasformazioni ambientali isotrope, con le quali però non è possibile uscire dall'insieme. Due configurazioni appartenenti allo stesso insieme sono dette equivalenti e gli insiemi di configurazioni equivalenti sono detti classi di equivalenza. Sono queste classi che si identificano con una certa tipologia di nodo. Si tenga presente che tra queste tipologie è annoverato anche l'unknot, ovvero la corda "snodata" (*Figura 1.1 (a)*).

È utile inoltre introdurre lo strumento più utilizzato per rappresentare un nodo: la sua proiezione. Come suggerisce il nome, questa rappresentazione riconduce il nodo all'immagine bidimensionale della varietà proiettata su un piano arbitrario. La proiezione di una curva di Jordan può, e spesso accade, non essere una curva di Jordan a sua volta e dunque presentare punti di autointersezione detti incroci. Questi punti sono

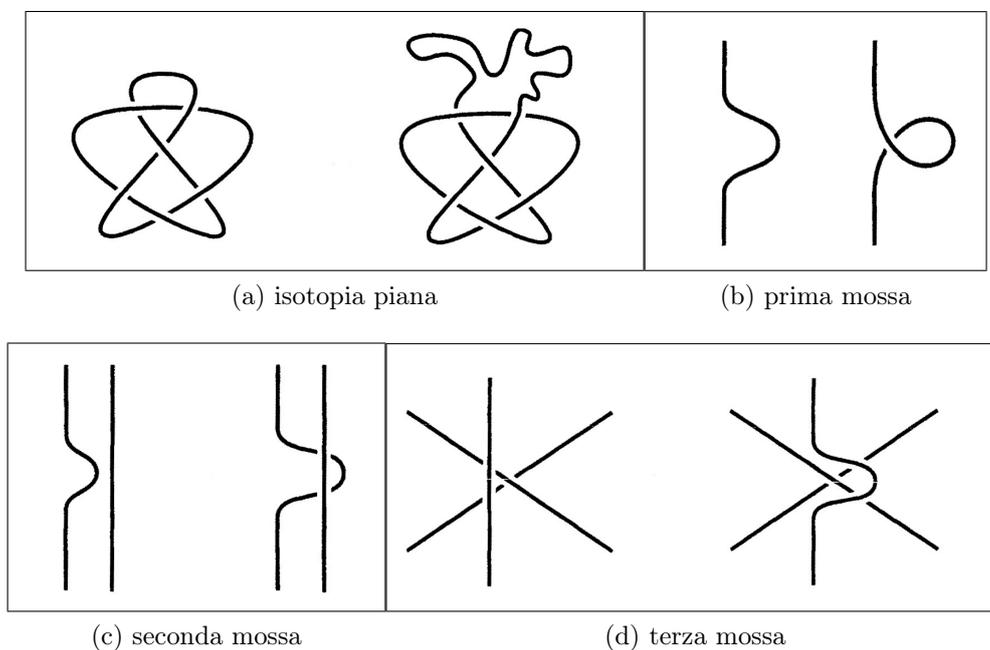


Figura 1.2: Esempi degli effetti delle trasformazioni descritte in precedenza. La terza mossa ha come scopo di trasformare un incrocio di tre segmenti in tre incroci di due segmenti. Per ottenere la gamma complessiva delle mosse di Reidemeister è necessario includere anche il corrispettivo delle presenti con i vertici sottosopra

d'importanza cruciale nella rappresentazione. Perché questa possa dirsi completa gli incroci devono esprimere in qualche modo anche l'informazione di quale dei due segmenti che si incrociano stia "sopra" e quale "sotto". Le immagini di *Figura 1.1* sono esempi di queste rappresentazioni. Per ovvi motivi è preferibile studiare il nodo utilizzando la sua proiezione e dunque sorge il bisogno di rendere in questa rappresentazione gli effetti delle trasformazioni ambientali isotopiche.

È dimostrato che qualsiasi trasformazione di questo tipo è rappresentabile nella proiezione attraverso una combinazione di isotopie piane e delle tre mosse di Reidemeister:

- Le isotopie piane sono, similmente a quelle ambientali, trasformazioni di tratti della corda che ne conservano la continuità e che soprattutto non cambiano il numero di incroci della rappresentazione.
- Le mosse di Reidemeister sono invece gli unici modi consentiti per rimuovere o aggiungere incroci senza snaturare la rappresentazione. Queste garantiscono che gli incroci su cui andiamo ad agire siano triviali e dunque che le proiezioni prima e dopo queste mosse rappresentino sempre nodi tra loro equivalenti.

Le isotopie piane e le mosse di Reidemeister sono tutti i modi in cui le trasformazioni isotopiche ambientali si manifestano nella nostra proiezione bidimensionale. Se ne osservi gli esempi in *Figura 1.2*. Gran parte della teoria dei nodi è basata su questa rappresentazione che contiene al suo interno tutte le informazioni utili riguardanti l'oggetto di studio. Dalla proiezione di un nodo è possibile ricavare, attraverso vari algoritmi, parametri invarianti per trasformazioni ambientali isotope. Questi invarianti sono utili per studiare le categorie di nodo. Il fatto che due nodi presentino un invariante differente ci assicura che questi non sono equivalenti, ma di norma non è possibile affermare il contrario.

Si pone a questo punto la questione di come ricondurre quest'apparato teorico a una varietà che presenti estremi. Intuitivamente in una corda limitata, se vogliamo verificare la presenza di un nodo, l'operazione più banale è quella di prendere gli estremi e allontanarli l'uno dall'altro, fino al punto in cui la corda non si tenda e non mostri, eventualmente, un annodamento. Se a questo punto risaldiamo tra loro gli estremi o li prolunghiamo all'infinito possiamo anche ricondurci alla classificazione pensata per varietà chiuse per identificare matematicamente il tipo di nodo. Nonostante queste due operazioni possano sembrare banali sono estremamente rilevanti non solo nella classificazione del nodo, ma anche nel suo sviluppo. È dunque fondamentale stabilire nel dettaglio come siano da compiersi. Per comprendere in che modo siano determinanti si prenda in considerazione il seguente caso. Si consideri un filamento in una configurazione qualsiasi; per rinsaldare i capi si può decidere di percorrere a ritroso la varietà con uno dei capi, rimanendole in ogni momento infinitesimamente vicino. Quando si sarà giunti all'altro estremo i due capi verranno uniti. Dal punto di vista matematico nessuna forza (attrito, rigidità) e nessun volume vengono presi in considerazione e dunque questa procedura non presenta problemi nemmeno se fosse presente sul filamento un nodo già stretto. È evidente però che se scegliamo un tale metodo per risaldare gli estremi l'unico nodo che otterremo sarà l'uniknot, a prescindere dalla condizione di partenza.

La matematica non è di alcun aiuto nello stabilire la procedura con cui vadano rinsaldati gli estremi e non resta che considerare la fisica del sistema che ci si trova a studiare. Dal punto di vista fisico una procedura come quella portata in esempio non è sempre praticabile, tant'è che risulta assai difficile anche nelle situazioni quotidiane sciogliere alcuni nodi che ci troviamo davanti. Si adotterà dunque una procedura più verosimile.

1.2.1 Procedura

Innanzitutto si impone tra gli estremi un forza di repulsione di una certa intensità. Può essere necessario lasciar trascorrere un certo lasso di tempo, ma questa forza spingerà gli estremi inevitabilmente in una posizione periferica del nostro sistema fino a che non intervenga la tensione della corda a bloccare il loro allontanamento. Un modo efficace per stabilire quando si sia concluso l'allontanamento potrebbe essere quello di aspettare fino a che la velocità di un estremo rispetto all'altro non sia nulla. Finché sussiste la tensione

tra gli estremi, questa configurazione si può considerare stabile. Occupa palesemente una posizione di minimo (non necessariamente assoluto) di potenziale rispetto alle altre a lei attigue, che in ogni caso prevedono un lavoro per contrastare la tensione. L'intenzione è, in virtù di questa stabilità, quella di fare di queste configurazioni il rappresentante delle nostre categorie di nodi su varietà aperte riconducendoli a una specifica classe di nodi su varietà chiuse.

Per compiere quest'altro passo si proietti la varietà su un piano qualsiasi, purché non sia quello esattamente perpendicolare alla congiungente tra gli estremi. In questa circostanza sarà sempre possibile unire le proiezioni dei due estremi in con una curva che non dia luogo a intersezioni né con se stessa né con la proiezione originale. Fatto ciò otteniamo una proiezione in tutto conforme a quelle richieste dall'apparato della teoria dei nodi e dunque siamo in grado di identificare il tipo di nodo.

La procedura illustrata non è certamente l'unica pensabile, ma è in grado di associare una qualsiasi configurazione di una varietà limitata a una categoria di nodo. In generale perché una procedura sia efficace è fondamentale che gli estremi siano portati in qualche modo in una posizione periferica rispetto agli altri punti della varietà, di modo che quando si va a proiettare su un piano sia possibile congiungerli senza creare incroci. È bene tener presente che, partendo dalla medesima configurazione, diverse procedure possono dare esiti diversi e dunque è necessario formularne una conforme ai propri interessi.

Si può senza dubbio affermare che la definizione di nodo su varietà limitata non possa prescindere dalla definizione di una procedura che lo riconduca a un nodo su varietà chiusa.

1.3 Proteine e nodi

Uno degli ambiti in cui lo studio delle proprietà non locali di varietà unidimensionali può trovare applicazione è la biochimica e in particolare lo studio delle proteine. Queste sono costituenti delle cellule e sono le responsabili di tutti i processi che avvengono al loro interno. Considerando la varietà di tipi di cellule diversi e la varietà di processi diversi che avvengono in ogni tipo di cellula ci si rende conto della quantità sterminata di proteine che sono interessanti per la biologia. È stato però possibile identificare una struttura comune a tutte loro. L'entità fondamentale di cui risultano essere costituite le proteine sono molecole dette aminoacidi. Tutti gli aminoacidi (o la maggior parte) hanno una struttura comune che si sviluppa attorno a un atomo di carbonio, detto *carbonio- α* . Il carbonio è in grado di effettuare quattro legami covalenti prima di diventare stabile, ma solo uno di questi quattro legami caratterizza l'aminoacido: gli altri legami sono gli stessi per tutti. Considerando una classica struttura a tetraedro di una molecola, in due dei vertici troviamo sempre un gruppo ammino(H_3N^+) e un gruppo carbossile(COO^-) legati in modo covalente. Con gruppo si intende una subunità della molecola o della proteina alla quale sono associate ben definite proprietà. A uno dei due restanti vertici

è legato un atomo di idrogeno e mentre nell'ultimo sito di legame troviamo il cosiddetto *side chain group*. È questa la parte che caratterizza gli aminoacidi e ne determina le differenti proprietà.

Gli aminoacidi che entrano in gioco nella composizione di proteine utili agli organismi viventi sono solo 21 a fronte della sterminata varietà di proteine che vanno a formare. Ogni proteina infatti è frutto della composizione di centinaia o addirittura migliaia di aminoacidi e ognuna di queste molecole ha nella proteina una posizione ben definita. Per tenere assieme tutti questi aminoacidi sono impiegati svariati tipi di legame, ma la struttura principale della proteina, il cosiddetto *Backbone*, è costituita da una successione di legami covalenti tra il gruppo carbossile di un aminoacido e il gruppo ammino di quello successivo. Il fatto che questi gruppi siano presenti in ogni aminoacido fa sì che si possano costituire catene molto lunghe di queste molecole con le più varie composizioni. Il filamento così costituito è soggetto a piegamenti e rotazioni attorno al proprio asse. Questa è quella che viene definita struttura primaria.

Una volta che viene costituita la struttura primaria entrano in gioco i gruppi funzionali specifici di ogni singolo aminoacido. Queste subunità interagiscono tra loro e con l'ambiente in cui si trova la proteina non attraverso legami covalenti, ma più debolmente. Ciò però è sufficiente per determinare il modo in cui si ripiega su sé stesso il *Backbone* dando luogo alla struttura secondaria. Gli esempi più comuni di queste strutture sono α -elica e β -foglietto. Proteine molto complesse arrivano a sviluppare strutture terziarie e addirittura si parla di strutture di quarto livello quando una proteina è composta da più subunità. Ai fini della nostra trattazione però tutte le strutture, le quali non coinvolgono legami covalenti, possono essere assimilate. Ogni proteina al termine del ripiegamento assume uno stato ben definito nel quale la posizione di tutti gli aminoacidi che la compongono è determinata. Questa configurazione è stabile solo in determinate condizioni di temperatura, PH e altri fattori che si verificano negli organismi che contengono le proteine.

Una proteina viene solitamente prodotta in un *ribosoma*. Questi organuli della cellula sono in grado di decodificare le informazioni contenute nel tRNA e di comporre strutture primarie delle proteine richieste dall'organismo. Non si occupano però della composizione della struttura secondaria (terziaria e di quarto livello): è infatti la proteina stessa che si ripiega autonomamente dando luogo al cosiddetto *foldiing*. Solo un determinato set di condizioni al contorno e l'esatta combinazione di aminoacidi determineranno un folding corretto.

Le proteine svolgono, come si è detto, molteplici funzioni, sia di tipo strutturale che chimiche e solitamente il loro ruolo è determinato dai *side chain group*. Questi gruppi però devono aver la possibilità di interagire con l'ambiente circostante. Non basta la loro presenza nella proteina ma è fondamentale che si trovino nella posizione giusta: esposti o interni, da un lato piuttosto che da un altro. Non è solo la sequenza di aminoacidi, ma anche la struttura secondaria (terziaria e di quarto livello) che determina le proprietà della proteina. Il processo di *foldiing* delle proteine risulta quindi cruciale per il corretto

funzionamento delle cellule e dell'organismo. Si possono distinguere le proteine in base alle tempistiche di questo processo: *two-step* sono quelle proteine che raggiungono la loro struttura secondaria (terziaria o di quarto livello) in un solo passaggio, mentre si dicono *multi step* quelle che ne impiegano più d'uno. Data l'importanza e la delicatezza della struttura secondaria (terziaria e di quarto livello) ci si è chiesti se questa potesse essere determinata o compromessa dalla presenza di nodi e più nello specifico se il fenomeno di nodo potesse essere connesso in qualche modo al fenomeno del *folding* e al numero delle sue fasi.

Per chiarire se si possa o meno rappresentare la proteina attraverso la simulazione è bene osservare altre sue rappresentazioni. Ne esistono di varia natura ma una particolarmente efficace ai nostri scopi è la mappa di distanza. Questa è costituita da una matrice quadrata di rango pari al numero di aminoacidi della proteina che si intende rappresentare. Il valore $i - j$ è la distanza tra l'aminoacido i e l'aminoacido j nella catena. Gli indici sono associati agli aminoacidi a seconda del loro ordine nel *backbone*. Come riferimento della posizione degli aminoacidi sono scelte le coordinate del carbonio- α . Il risultato di questa rappresentazione è riportato in *Figura 1.3(a)*. Considerando che la distanza per lo sviluppo di un legame tra due aminoacidi è di 8\AA , si può ottenere dalla mappa di distanza, la mappa di contatto. Riportata in *Figura 1.3(b)*, la mappa di contatto mette in evidenza solo le caselle con un valore inferiore alla soglia di legame. Da questa rappresentazione è evidente che la maggior parte dei legami è vicino alla diagonale e costituisce il *backbone*. La mappa di contatto non ci da informazioni sulla struttura secondaria della proteina se non in modo parziale. Sono comunque sufficienti per convincerci della continuità del *backbone* in tutta la proteina e del fatto che i legami tra aminoacidi non consecutivi siano in netta minoranza. I dati sulle mappe di distanza e di contatto sono stati presi da [6].

Sulla base di queste considerazioni si è pensato di applicare la simulazione sviluppata allo studio delle proteine. Le proprietà del filamento simulato riproducono molto fedelmente le dinamiche della struttura primaria delle proteine, compresa la possibilità di rotazione attorno all'asse del legame. La distanza tra un aminoacido e quello successivo è costante visto che si tratta di legami dello stesso tipo. Sono però ignorate le differenti proprietà chimiche del *Side chain group* e le differenti masse. Dunque gli aminoacidi vengono considerati tutti uguali e tra loro interagenti solo attraverso urti e la tensione del filamento e sono trascurati tutti i legami che non fanno parte del *backbone*.

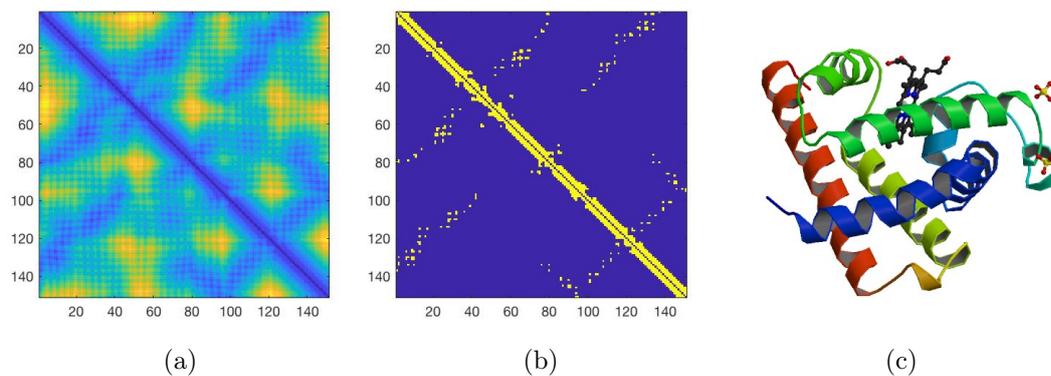


Figura 1.3: L'immagine (a) rappresenta la mappa di distanza della proteina 1A6N. Le gradazioni di colore vanno dal blu al giallo con l'aumentare della distanza. Nell'immagine (b) vediamo la rispettiva mappa di contatto dove, in giallo, sono evidenziati tutte le distanze inferiori agli 8\AA . In fine l'immagine (c) contiene la rappresentazione 3D della stessa proteina. Le immagini (a) e (b) sono state prese da [6]. L'immagine (c) è presa dalla [4]

Capitolo 2

Simulazione

Sulla base di un modello fornito dal professor Sandro Rambaldi del dipartimento di fisica e astronomia di Bologna, si è sviluppata una simulazione in grado di rappresentare, sotto certe condizioni, un filamento. Ciò ha reso possibile mettere in opera quanto esposto riguardo ai nodi su varietà con estremi. È stato utilizzato il linguaggio C++ per lo sviluppo. Ultimata la parte principale di tale programma, la quale consta circa di 700 linee di codice, se ne sono ricavate tre versioni differenti. Due sono state impiegate per i test su toy-model e una è stata utilizzata, attraverso uno script, per esaminare il database di proteine.

2.1 Funzionamento della simulazione

Il calcolo della simulazione è basato sull'oggetto della classe creata ad hoc `Vector9D`, rappresentativo di un punto materiale di massa unitaria. Questi oggetti saranno chiamati "punti", senza alludere alla natura adimensionale dell'omonimo oggetto matematico. Questo tipo di oggetti contiene solo 3 metodi, 3 array tridimensionali che rappresentano le grandezze vettoriali di interesse di un punto materiale di massa unitaria: posizione, velocità e accelerazione. La simulazione può gestire una quantità arbitraria di punti anche se questo comporta un aumento della mole di calcolo e dunque un rallentamento della simulazione.

I punti vengono creati all'avvio della simulazione e i loro metodi sono inizializzati dalla funzione `Start`. È fondamentale che la distanza tra due punti adiacenti sia, già all'avvio, unitaria. Si sono predisposte varie opzioni di avvio: una configurazione a spirale, una stesa e una con i punti in posizione casuale (sempre facendo attenzione a collocarli alla giusta distanza tra loro). Si è anche studiata una configurazione che, utilizzando la procedura descritta nel capitolo 1.2 da luogo a un nodo a trifoglio. Infine si è predisposta una funzione di input da file di testo per altre configurazioni di partenza e la si è resa compatibile con i file da cui è costituito il database di proteine studiato.

Creati tutti i punti il programma entra in un ciclo che si interrompe solo al termine della simulazione. Ogni iterazione comincia con il calcolo delle forze agenti su ogni punto a un certo istante. Di questo calcolo si occupano in due fasi le funzioni *ForzeEsterne* e *Forze*. *ForzeEsterne* si occupa delle forze derivanti dalle condizioni al contorno e da parametri strutturali di vario tipo:

- È simulata una rigidità del filamento in base a un parametro variabile. Maggiore è il valore di questo parametro, maggiore è la forza che il punto i subisce. Questa forza è ortogonale alla congiungente tra i punti $i - 1$ e $i + 1$ e rivolta verso di essa.
- Su ogni punto sono simulate stimolazioni in direzioni casuali per rappresentare l'agitazione termica di un mezzo in cui eventualmente potrebbe essere immerso il filamento. L'intensità media di queste simulazioni è variabile. Per ottenere effettivamente una distribuzione casuale delle stimolazioni si utilizza un generatore di numeri pseudo casuali fornito dalla *Standard Template Library*, *ranlux24_base*, il quale riceve il seme dal random device della macchina in uso.
- Una forza opposta alla velocità dei punti e ad essa proporzionale simula una viscosità dello spazio circostante al filamento. La proporzione tra forza e velocità è variabile.

Forze si occupa di sviluppare la tensione che ogni punto sente su sé stesso da parte del resto del filamento. Nessun parametro di questa funzione è variabile.

Si è scelto di separare queste due funzioni perché solo *Forze* è davvero indispensabile al funzionamento della simulazione: senza di essa non è possibile far muovere i punti in modo tale che conservino la continuità della varietà che rappresentano. *ForzeEsterne* è di natura accessoria e ha lo scopo di rendere la simulazione più pertinente a situazioni reali introducendo caratteristiche e parametri variabili all'avvio.

Una volta calcolate le forze e dunque le accelerazioni a cui ogni punto è soggetto in un certo istante vengono integrate le equazioni del moto per ottenere le coordinate e gli impulsi del sistema all'istante successivo. Ciò è svolto dall'algoritmo Runge Kutta 4 (RK4). Si tratta di un metodo di integrazione numerica delle equazioni differenziali del tipo

$$\frac{dx}{dt} = f(x, t)$$

In tal modo, nota la condizione iniziale (velocità e posizione) e le forze agenti a un certo istante su ogni punto del filamento, si è in grado di ottenere posizione e velocità dei punti all'istante successivo. È chiaro che, per essere perfettamente esatto, quest'algoritmo dovrebbe allontanarsi dalla condizione iniziale di un tempo infinitesimo prima di ricalcolare le forze su ogni punto. Ciò computazionalmente è impossibile e siamo dunque costretti a commettere un errore. Quest'algoritmo commette un errore che è legato all'intervallo DT tra i due istanti, in particolare cresce come DT^4 . Test su toy-model annodati,

messi in tensione con forze diverse, hanno anche permesso di impostare nella funzione che gestisce RK4 un DT adatto agli scopi della nostra simulazione. Si è scelto un valore abbastanza piccolo per ottenere una precisione soddisfacente per la nostra analisi, ma sufficiente a far evolvere la simulazione con un numero gestibile di iterazioni. Si tratterà di questo più diffusamente nel paragrafo 2.2.2.

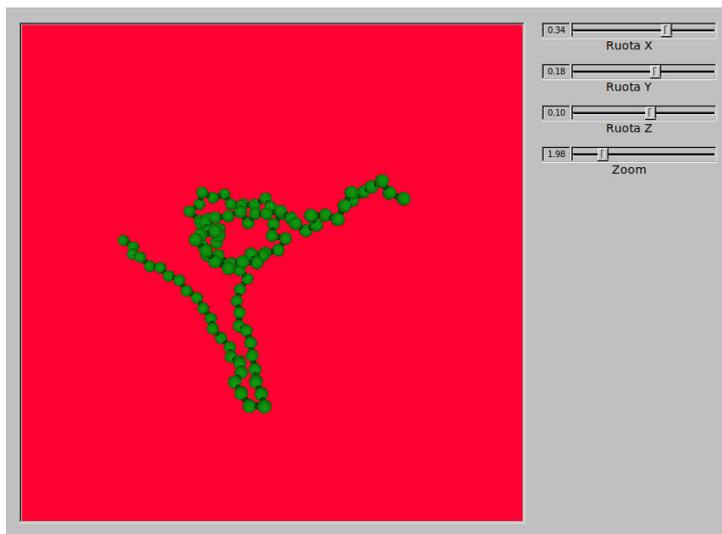


Figura 2.1: Si riporta la rappresentazione grafica della simulazione. Sono presenti gli slider per regolare zoom e angolo di visuale.

risultano più vicini di una certa soglia e in rotta di collisione, viene attivata *SviluppaUrti* che dà luogo all'urto applicando la conservazione della quantità di moto e dell'energia per calcolare le velocità successive all'urto. La soglia di distanza sotto la quale risulta positivo il primo controllo di *ControllaUrti* rappresenta il raggio dei punti. In base a questo meccanismo i punti acquistano a tutti gli effetti un volume sferico e il filamento inizia ad assomigliare ad un oggetto fisico. Motivo per cui gli urti vengono gestiti separatamente da RK4 è che in questo calcolo viene utilizzata la conservazione della quantità di moto e dell'energia per calcolare gli impulsi, e non un'equazione di campo come quelle richieste da RK4. terminate queste operazioni si conclude un'iterazione del ciclo. La simulazione può concludersi e generare l'output o proseguire ricominciando da *ForzeEsterne*.

Data la mole di dati da gestire parte fondamentale del programma è costituita dalla rappresentazione grafica del filamento in uno spazio tridimensionale. Avere a che fare con un fenomeno non lontano dall'esperienza quotidiana rende l'occhio stesso uno strumento di verifica efficace. Per esempio, prima che il DT di RK4 fosse adattato, rendersi conto che il filamento era soggetto a compenetrazione sarebbe stato quasi impossibile osservando esclusivamente colonne di numeri. Per la grafica sono state utilizzate le librerie

Dopo ogni iterazione di RK4 vengono eseguite anche le funzioni *ControllaUrti* e *SviluppaUrti*. Queste hanno lo scopo di gestire gli urti elastici tra due punti che si trovano ad essere a contatto. La prima funzione esegue due controlli su tutti i punti. Per prima cosa calcola la distanza tra il punto i e tutti gli altri punti. Se questa distanza è minore di una certa soglia per un punto j , si esegue anche un controllo sulle velocità dei due punti. Si calcola la componente parallela alla congiungente $i \rightarrow j$ della velocità relativa di j rispetto a i . Se questa componente è negativa vuol dire che i due punti stanno andando uno verso l'altro. Se i punti ri-

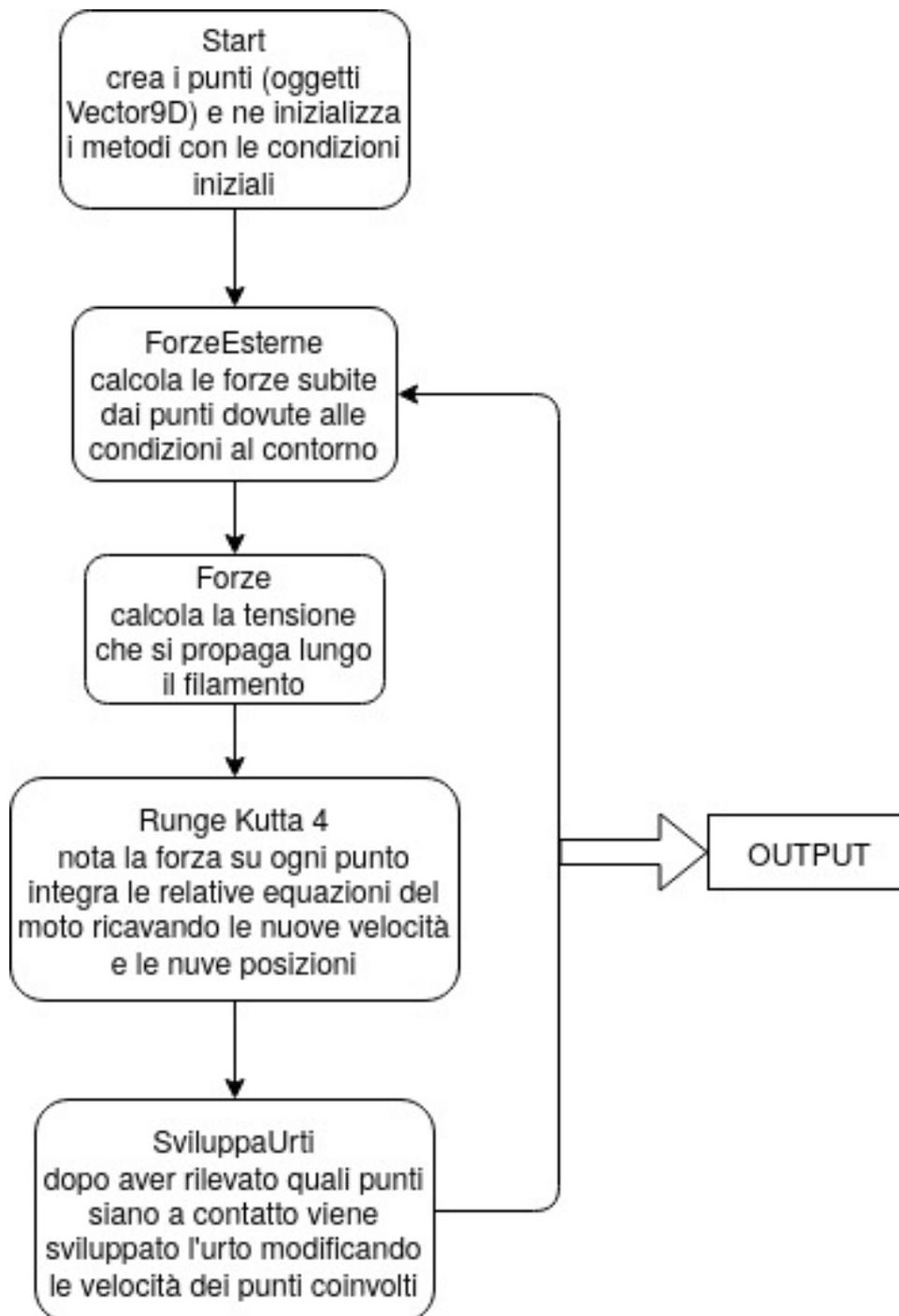


Figura 2.2: Si rappresenta in modo schematico il funzionamento del programma

FLTK e OpenGL. La prima è servita a gestire lo zoom e la rotazione del punto vista attorno ai 3 assi in tempo reale; la seconda ha permesso di rappresentare la posizione degli oggetti vector9D attraverso l'oggetto SolidSphere e gestire l'animazione che segue l'evoluzione della simulazione. Il risultato è mostrato in *Figura 2.1*.

Si riassume schematicamente il funzionamento descritto in *Figura 2.2*

2.1.1 Limiti

È importante tener presente che questa simulazione, per quanto realistica, presenta dei limiti. Il primo è il fatto che il solido che viene simulato non è perfettamente cilindrico, ma ha le fattezze di una successione di sfere molto vicine. Quando il sistema sfrega su se stesso genera dunque un attrito che può diventare molto elevato. Questo risulta di particolare intralcio quando si desidera stendere il sistema applicando una tensione agli estremi: può succedere infatti che l'attrito del filamento sviluppato in certi punti particolarmente aggrovigliati bilanci questa tensione bloccando lo stiramento. Altro limite è il fatto che il filamento simulato non oppone alcuna resistenza alla torsione sul proprio asse. Alcuni sistemi reali, come proteine costituite di aminoacidi legati da legame singolo, presentano la stessa caratteristica ma altri, come una qualsiasi corda, no.

Oltre a vincoli di natura intrinseca dovuti alla logica di programmazione ve ne sono altri relativi alla limitatezza della potenza di calcolo a disposizione. Il primo riguarda il fatto che per garantire la non compenetrazione si è fissata una precisa dipendenza tra due parametri geometrici del nostro sistema. La distanza tra il centro di due punti tende ad essere doppio del raggio dei punti. Questo minimizza la discontinuità del filamento senza dar luogo ad attivazioni erranee di *ControllaUrti*. Il cilindro, al quale si può approssimare la successione di punti che gestisce il programma, ha dunque le seguenti misure:

$$\text{Raggio} = R \quad \text{Altezza}(H) = N * R \quad H/R = N$$

dove N è il numero di punti gestiti. Per simulare filamenti più affusolati e quindi incrementare il rapporto H/R abbiamo come unica opzione quella di aumentare N . La gestione di un maggior numero di punti però richiede maggiori tempi di calcolo e dunque filamenti più affusolati risultano più difficilmente simulabili.

Sempre per evitare la compenetrazione siamo costretti a inserire anche un vincolo relativo all'intensità delle stimolazioni in cui incorre il sistema. Questo vincolo è imposto sull'accelerazione massima alla quale possono essere soggetti i punti. Considerando

$$F = M * a$$

È facile accorgersi che ciò vuol dire imporre una limitazione sulle forze agenti sul sistema attraverso la costante arbitraria della massa (M) del singolo punto. Sistemi più leggeri sono dunque più esposti alla compenetrazione a parità di forze simulate di sistemi pesanti. Come si approfondirà in seguito, questo limite è strettamente legato al parametro DT della funzione RK4.

2.2 Risultati dello stiramento di toy-model

Conclusa la parte di sviluppo della simulazione è stato necessario testare la sua l'efficacia e le sue potenzialità. Prima incognita da sciogliere è stato il rapporto tra il numero di punti materiali che il programma si può trovare a gestire e la velocità di calcolo. Come si è detto nel paragrafo 2.1.1, questo aspetto ha ripercussioni anche a livello geometrico sul sistema, determinando quanto simulare sistemi più affusolati sia più dispendioso in termini di tempo. Si prevedeva ovviamente che un aumento della mole di dati da gestire avrebbe portato a un aumento del tempo di calcolo. Ci si aspetta dunque che il tempo necessario per una singola iterazione del ciclo di azioni di cui consta la simulazione sia funzione strettamente crescente del numero di punti gestiti. Si sono dunque fatti dei test con simulazioni aventi diverso numero di punti.

Un altro fenomeno che ha richiesto uno studio attraverso toy-model è quello della compenetrazione del filamento. La natura sfuggente di questo aspetto del sistema ha reso complicato anche solo rendersi conto della sua esistenza. Ciò non di meno è evidente che se si verifica una compenetrazione durante una simulazione, questa perde di attinenza con quanto esposto riguardo alla teoria dei nodi. L'origine di questo fenomeno è stata identificata nel fatto che la gestione degli urti tra i punti è separata dall'azione di RK4. Accorpare le due funzioni avrebbe incrementato drasticamente la mole di calcolo costringendo a trattare i punti come campi di forza repulsivi piuttosto che sfere rigide. Si è dunque deciso di non modificare il funzionamento della simulazione e piuttosto di capire in quali circostanze la compenetrazione si verificasse e in quali fosse possibile escluderla.

2.2.1 Rapporto tra numero di punti e tempo di calcolo

Come già accennato l'avanzamento della simulazione è scandito dal ripetersi delle iterazioni di un ciclo. Ogni iterazione dà luogo a svariate operazioni, la maggior parte delle quali va compiuta una volta per ogni punto per calcolarne i parametri. In un sistema di entità non interagenti, o interagenti solo a corto raggio, osserveremmo sicuramente una dipendenza del tempo di calcolo lineare rispetto al numero di entità. Nel sistema in esame però la tensione lungo il filamento si propaga istantaneamente (a ogni azione di *Forze*) attraverso tutti i punti. Questa è a tutti gli effetti un'interazione a lungo raggio e non può che dar luogo a una componente quadratica nella dipendenza tra il tempo per l'esecuzione di una singola iterazione del programma (τ) e il numero di punti gestiti (N).

Si sono dunque fatti vari test utilizzando toy-model contenenti diverso numero di punti e si è misurato il tempo di calcolo attraverso la funzione *clock()* della libreria *CTime*. Sulla base dell'ipotesi di cui sopra si è provato un fit di questi dati utilizzando una funzione polinomiale di secondo grado. È stato eliminato il parametro costante dalla funzione di prova perché si è osservato che, in assenza di punti, il programma non compie nessuna operazione tra un'iterazione e il χ quadro risulta comunque soddisfacente. Si è

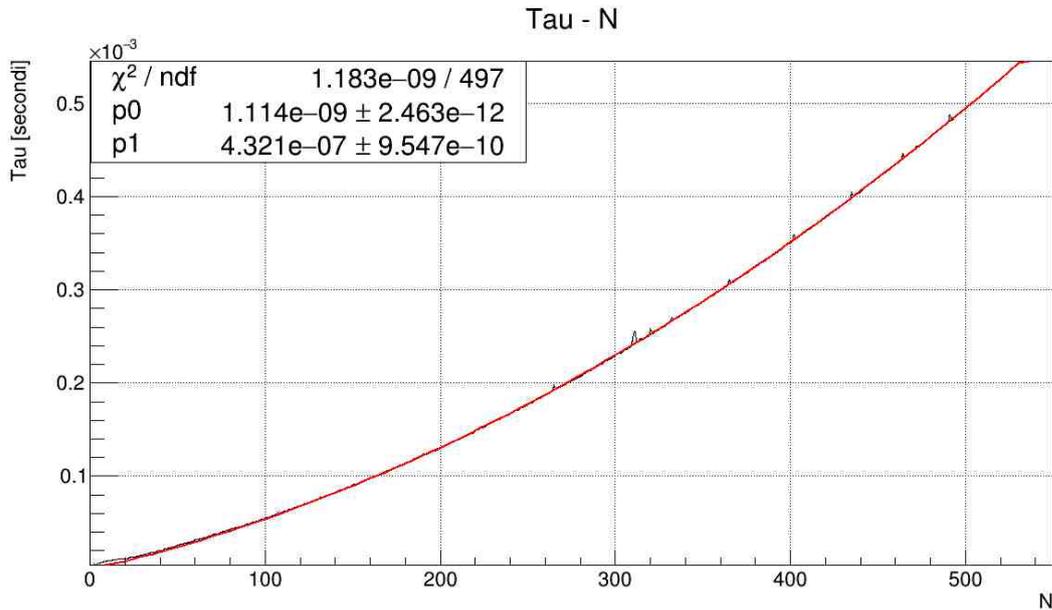


Figura 2.3: Grafico dei dati di tempo per numero di punti. Al fine di risentire in modo minore possibile di interferenze nella singola iterazione si è misurato il tempo necessario a 10000 iterazioni e qui si riporta il tempo medio stimato per la singola. Sono present anche i valori dei parametri ricavati dal fit.

giunti al seguente guess per la funzione $\tau(N)$

$$\tau(N) = p0 * N^2 + p1 * N$$

Dove $p0$ e $p1$ sono parametri liberi. È bene tenere presente che i dati raccolti sono specifici della macchina utilizzata per il calcolo e non si esclude che su diverse macchine possano avere valore diverso. Per confrontare la velocità di calcolo con il tempo simulato si è studiato il parametro $\tau(N)/DT$. Ogni iterazione impiega $\tau(N)$ secondi ad essere calcolata e fa progredire la simulazione di DT . Sarebbe ovviamente auspicabile mantenere questo parametro minore di 1, ma non sempre risulta possibile per N molto grandi.

Si riportano i risultati nel grafico in *Figura 2.3*. In nero sono riportati i tempi impiegati per un'iterazione del ciclo del programma in relazione ai punti gestiti. Per avere dati più precisi possibile si sono eseguite non una singola iterazione, ma 10000 per ogni campionamento. Le piccole irregolarità che si possono osservare sono probabilmente dovute a processi paralleli che hanno impiegato la CPU mentre veniva eseguita la simulazione. In rosso in sovrapposizione è riportata la curva identificata come best fit. Sia per il fit che per la produzione del grafico si è usato ROOT cern.

Sia dal grafico che dal valore del χ quadro ridotto è evidente che il fit sia attendibile. Il valore dei parametri ottenuti evidenzia una dipendenza quadratica molto marcata del tempo di calcolo dal numero di punti gestiti.

$$p0 = 1.114 * 10^{-9} \pm 2.463 * 10^{-12}s \quad p1 = 4.321 * 10^{-7} \pm 9.547 * 10^{-10}s$$

Nel database erano presenti proteine di svariata lunghezza, da una cinquantina a più di 4000 aminoacidi. Si è dovuto escludere quelle più lunghe visto che i tempi di elaborazione sarebbero stati ingestibili.

2.2.2 Limite di non compenetrazione

Il primo passo per studiare il fenomeno della compenetrazione è stato riuscire a rilevarlo.

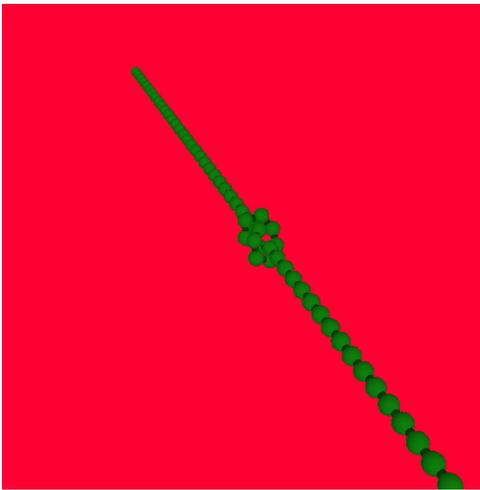


Figura 2.4: Si osserva un comune nodo a trifoglio in tensione. È questo il sistema su cui si è testata la compenetrazione. L'immagine è ricavata dalla grafica della simulazione

Non si è riusciti a trovare nessun algoritmo in grado di metterlo in luce utilizzando esclusivamente le coordinate dei punti di una simulazione durante una sua esecuzione. Questo del resto è il motivo per il quale non si è in grado di escludere la compenetrazione implementano una funzione nel programma ad hoc. Solo in certe configurazioni è possibile osservare gli effetti di questo fenomeno in maniera chiara: una di queste è quando il filamento sviluppa un nodo a trifoglio (il suo corrispondente su una varietà non chiusa) e gli estremi sono messi in tensione. Se questo nodo si trova in tensione e si scioglie vuol dire che è avvenuta una compenetrazione. Per studiare questo fenomeno si è dunque scelto di osservare gli effetti di diverse combinazioni di parametri su un sistema avente questa configurazione. In *Figura 2.4* possiamo osservare la configurazione del sistema usato per le simulazioni.

Il dato che si è utilizzato per sancire l'avvenuta compenetrazione è la distanza tra gli estremi. In questa configurazione, prima della compenetrazione questa distanza è inferiore di svariate unità (diametro del punto) alla lunghezza del filamento, a causa della presenza nodo. Quando però avviene la compenetrazione il nodo si scioglie e la distanza tra gli estremi, soggetti a tensione, eguaglia in breve tempo la lunghezza del filamento.

Grazie a questo espediente si è potuta valutare la correlazione di vari parametri del programma con tale fenomeno. Si è constatato che la compenetrazione si verificava prima per valori elevati di tensione tra gli estremi e per valori elevati del parametro DT della funzione RK4. Ciò non di meno, anche per bassi valori di entrambi la compenetrazione ha comunque luogo. È però necessario eseguire un maggior numero di iterazioni del ciclo

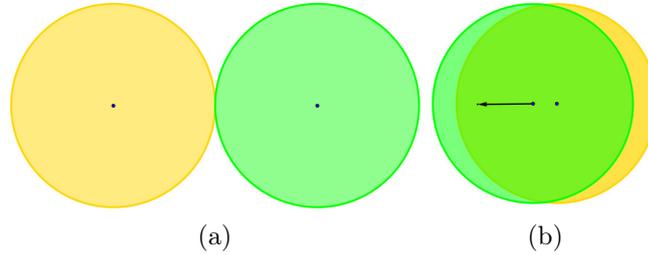


Figura 2.5: Rappresentazione di due punti prima (a) e dopo (b) l'azione di RK4 in un caso che dà luogo a una compenetrazione. Si considera il fenomeno dal sistema di riferimento del punto giallo. Questa situazione può verificarsi se l'accelerazione del punto verde è $4R/DT^2$

del programma, durante le quali si mantenga applicata la tensione. Si cerca dunque di trattare la compenetrazione da un punto di vista statistico. L'obiettivo di questa trattazione è associare una certa distribuzione di probabilità di compenetrazione a un set di parametri. Questa distribuzione sarà riferita al numero di iterazioni del programma e dunque anche al tempo simulato.

Alla luce delle precedenti osservazioni si è anche formulata un'ipotesi per spiegare questo fenomeno. Qualora due punti siano tenuti a contatto e tra i due si sviluppi una forza (attrattiva) superiore a una certa soglia, può accadere che, dopo una singola azione di RK4, il centro di uno dei due si trovi a passare oltre il centro dell'altro. Ciò avviene in virtù della forza a cui i punti sono sottoposti, la quale sviluppa una tale accelerazione da far attraversare loro lo spazio necessario in un solo DT . Quando ciò accade, *ControllaUrti* non segnala alcun urto tra i due punti. Questo accade perché il controllo effettuato da questa funzione non è relativo solo alla distanza tra i centri dei punti, ma anche alla loro velocità: l'urto ha luogo solo se la velocità relativa di uno dei due è verso il centro dell'altro nel momento in cui *ControllaUrti* viene eseguita. Se si verifica quanto sopra, la condizione sulla velocità dei due punti a contatto viene ovviamente meno e dunque non si sviluppa alcun urto. Ciò fa sì che avvenga la compenetrazione. Si riporta una rappresentazione grafica di quanto scritto in *Figura 2.5*

Data la complessità delle interazioni d'urto tra i punti all'interno del nodo e l'irregolarità del diametro del nostro sistema non si è in grado di stabilire una soglia esatta sopra la quale si verifichi la compenetrazione. Questo fa sì che si debba scegliere un approccio statistico al fenomeno cercando di ridurre la probabilità che questo si verifichi. Si è scelto di associare a questo fenomeno una distribuzione poissoniana, fissato un certo set di parametri. Visto che il nodo risulta centrale nell'ambito di questo studio si è scelto di ricavare il parametro di questa distribuzione proprio dalla simulazione di un toymodel annodato. Questo nodo, già stretto in partenza, viene messo in tensione fino a che non si verifica la compenetrazione e il nodo si scioglie. Il dato utilizzato per rilevare la compenetrazione è la distanza tra gli estremi.

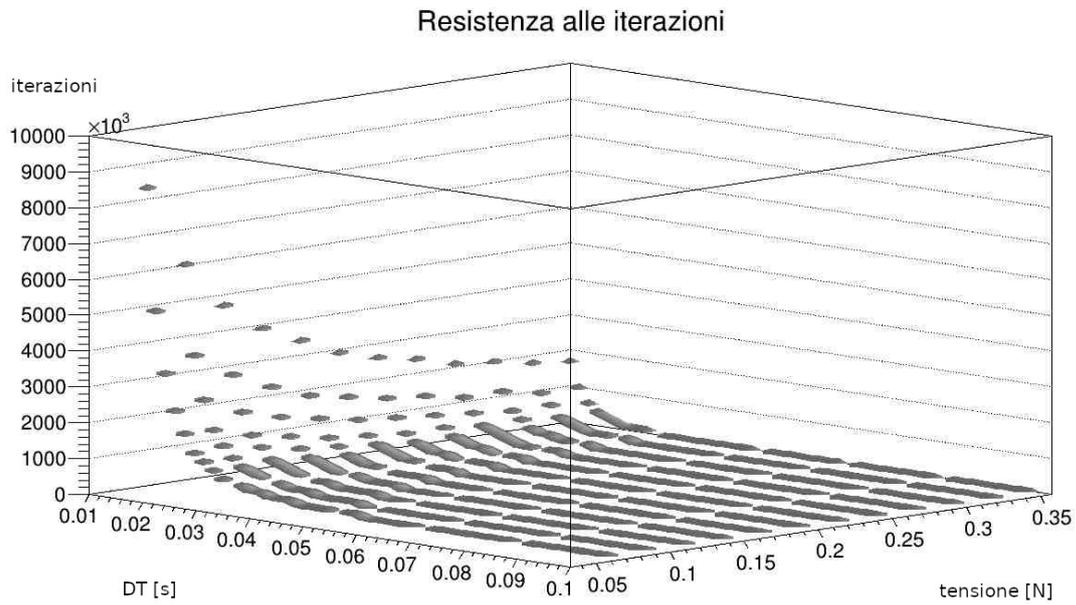
Il dato che risulta da questo studio è il numero di iterazioni che il programma compie prima che avvenga una compenetrazione. Questo dato corrisponde alla media della poissoniana relativa al set di parametri utilizzato, e anche alla sua deviazione standard. In tal modo fissiamo l'unico parametro libero di questa distribuzione e otteniamo una stima della probabilità che, in circostanze diverse da quelle simulate, con lo stesso set di parametri, si verifichi una compenetrazione prima di una data iterazione. Si noti che la configurazione simulata per ricavare questo parametro non è in realtà una configurazione qualsiasi, ma è probabilmente quella che più espone il sistema al fenomeno di compenetrazione. Per quanto riguarda le proteine non è detto che fin dall'inizio dello stiramento sia presente sulla varietà un nodo già stretto e soprattutto quando il sistema entra in tensione, come vedremo, si può di norma considerare lo stiramento finito. Fissare dunque la media della poissoniana sul dato raccolto da un toy-model annodato e in tensione per tutta la sua simulazione vuol dire compiere una stima per eccesso della probabilità di compenetrazione prima di un dato istante.

A questo punto si ricercato un set di parametri tale da ridurre la probabilità che avvenga una compenetrazione tra un'iterazione del programma e quella successiva. Se l'ipotesi formulata in precedenza è corretta i parametri di maggior influenza su questo fenomeno dovrebbero essere:

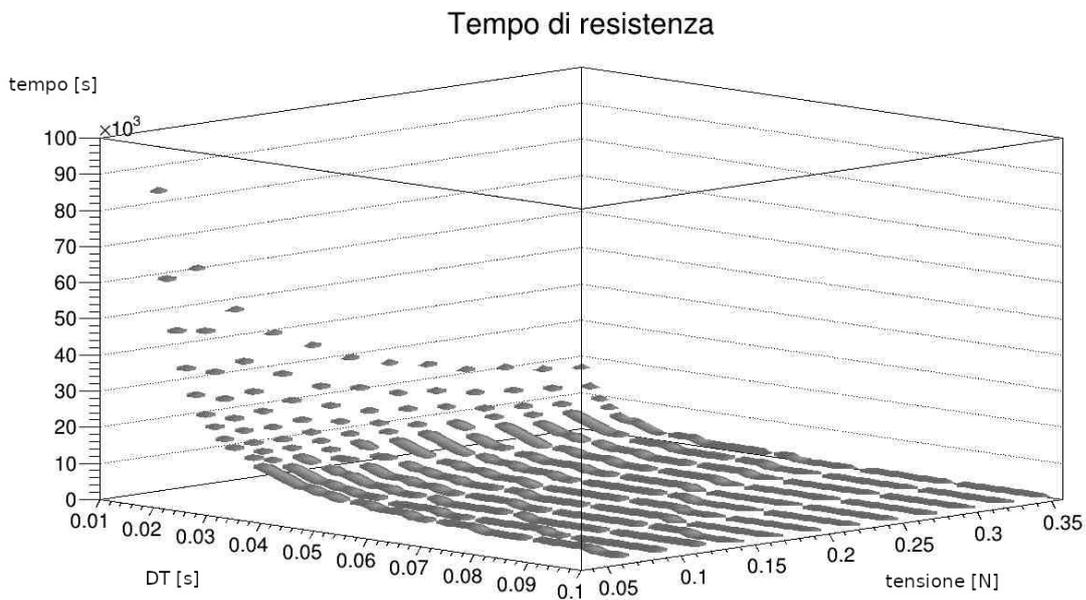
- La forza di trazione degli estremi,
la quale influenza indirettamente la forza tra i punti a contatto e dunque la loro accelerazione nel nodo e dovrebbe essere direttamente proporzionale alla probabilità.
- Il parametro DT della funizione RK4
che determina ogni quanto spesso entri in azione `ControllaUrti` e dunque dovrebbe essere direttamente proporzionale alla probabilità
- Il raggio dei punti
che influenza direttamente la distanza che i centri dei due punti devono compiere prima dell'azione di `ControllaUrti` per dar luogo alla compenetrazione, e dunque dovrebbe essere inversamente proporzionale alla probabilità.

Si è scelto di concentrare l'attenzione sui primi due parametri, i quali potevano essere modificati senza alterare eccessivamente le funzioni della simulazione. Si è trascurato il raggio dei punti. Aumentarlo oltre la metà della distanza tra i centri di due punti attigui avrebbe implicato risolvere il problema dello sviluppo di urti tra questi punti. Si noti che se si aumentassero anche le distanze punto si otterrebbe solo un riscaldamento del sistema in tutta la sua geometria.

Come si può notare dal grafico in *Figura 2.6* le previsioni sono confermate. Notiamo che il parametro che influisce più marcatamente sulla compenetrazione è DT . L'intensità



(a)



(b)

Figura 2.6: Il grafico *a* riporta il numero di iterazioni del ciclo del programma necessarie a far sì che il nodo si sciolga per una certa combinazione dei parametri di di *tensione* e *DT*. Il grafico *b* riporta il tempo necessario a far sì che il nodo si sciolga per una combinazione degli stessi parametri. I due dati sono dipendenti l'uno dall'altro in base al parametro *DT*. In entrambi e le acquisizioni si sono utilizzati punti di massa unitaria.

di tensione applicata ha un effetto rilevante solo se DT ha un valore abbastanza ridotto. Si riportano in *Figura 2.6 (b)* i valori di tempo simulato in cui il segmento ha resistito all'annodamento. I valori di questo secondo grafico corrispondono al numero di azioni di RK4 moltiplicato per il parametro DT . In questo grafico l'incremento del valore di tempo pre-compenetrazione rispetto a DT è meno marcato, ma comunque evidente. Si faccia attenzione alla diversa scala di grandezza utilizzata per l'asse verticale dei due grafici.

Si sceglie come coppia di parametri da utilizzare per lo stiramento delle proteine quella che ha portato il sistema a resistere alla compenetrazione per un maggior numero di iterazioni:

$$DT = 0.01s \quad Tensione = 0.05N$$

Fissati questi parametri si ottiene il parametro della poissoniana relativa alla probabilità di compenetrazione nel tempo.

$$Tm = 83518s$$

Questo parametro stabilisce la media della distribuzione e la sua deviazione standard. Il valore di tensione riportato è relativo a un sistema di punti di massa unitaria. Per mantenere lo stesso parametro Tm è bene riscaldare per la massa degli oggetti simulati il valore di tensione. Questo perchè il fenomeno di compenetrazione dipende dall'accelerazione dei punti, non dalla forza applicata su essi.

Capitolo 3

Risultati

Per studiare potenzialità e limiti della simulazione si è voluto metterla alla prova anche con dati non prodotti ad hoc. La scelta è ricaduta sulle proteine che sono ben rappresentate nelle loro proprietà dinamiche dalla simulazione. Il professor Daniel Remondini del dipartimento di fisica e astronomia di bologna ha fornito un elenco di proteine le quali erano già state catalogate in precedenza in base alle modalità di folding: *two-step* o *multi step*. Si è parlato di questa distinzione nella sezione 1.3. Si è deciso quindi di valutare una possibile correlazione tra il tipo di folding e la presenza di nodi.

Il database fornito contiene proteine di lunghezza molto varia: si riporta in *Figura 2.7* un grafico contenente la distribuzione delle proteine in base al numero di aminoacidi di cui sono costituite. Per ogni proteina è stato reperito dalla *Protein Data Bank* [4] il relativo file *.pdb*. Questi file, tra le altre informazioni riguardo alla proteina, contengono anche le coordinate spaziali del carbonio α di ogni aminoacido. Si sono dunque eseguite simulazioni nelle quali la configurazione iniziale del sistema era costituita da un sufficiente numero di punti inizializzati con le coordinate di questo atomo e velocità nulla. Si è ottenuto in questo modo un modello tridimensionale della proteina, nella quale tutti gli aminoacidi sono rappresentati da un punto. Il vincolo di distanza reciproca tra i punti successivi nel filamento simulato rappresenta il legame covalente che si sviluppa tra due aminoacidi attigui nel *backbone*. Come già accennato nel capitolo 1.3, non sono presi in considerazione gli altri legami della proteina.

Rappresentata la proteina nella configurazione iniziale, si è messa in pratica la procedura descritta nel paragrafo 2.1.2. Questo processo verrà di seguito chiamato *stiramento*. Si è applicata una tensione costante agli estremi del filamento parallela alla congiungente di tali punti. Raggiunta una condizione di stabilità del sistema in tensione si è predisposto affinché la simulazione si interrompesse. È stata stabilita come condizione per concludere lo stiramento l'uguaglianza delle velocità dei due estremi. Non si impone un'uguaglianza esatta tra le due velocità, ma che la loro differenza sia sotto una certa soglia. Esclusi i primissimi istanti della simulazione, nei quali non viene applicato il controllo, le due velocità si eguagliano solo se il sistema entra in tensione. Per le simu-

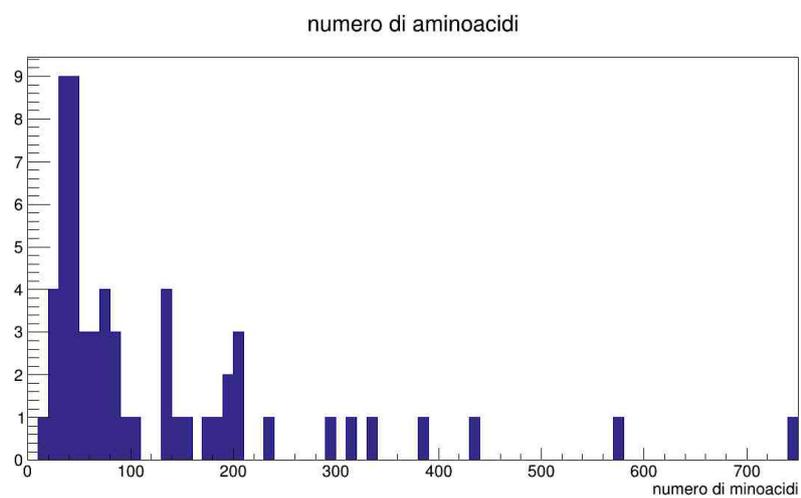


Figura 3.1: Nell'istogramma è riportata la distribuzione delle proteine in base al numero di aminoacidi che le compongono

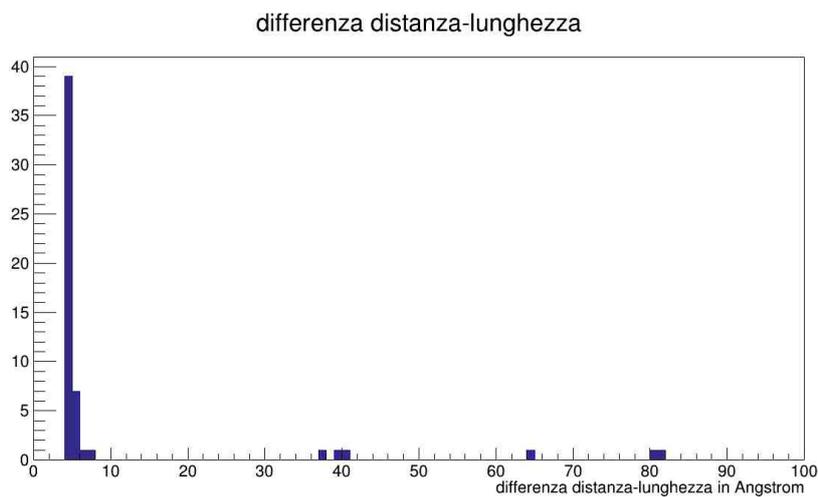


Figura 3.2: Nell'istogramma è riportata la distribuzione delle differenze tra lunghezza della proteina e distanza tra gli estremi al termine dello stiramento

lazioni si sono utilizzati parametri che minimizzassero la possibilità di compenetrazione: $DT = 0.005$ s e forza di repulsione = $8.3 * 10^{-27}$ N. Si è riscalata la forza applicata nel toy-model della sezione 2.2.2 per la media della massa degli aminoacidi in modo da ottenere la stessa accelerazione. Sono state escluse dallo studio tre proteine: 1AON di 4008 aminoacidi, 2ABD di 1247 aminoacidi, 2PTL di 819 aminoacidi. Il programma non è stato in grado di gestire in tempi ragionevoli lo stiramento di queste proteine, nei primi due casi non è stato nemmeno possibile dare inizio alla simulazione.

Dal filamento in tensione si ricava il dato della distanza tra gli estremi. Come già accennato nel paragrafo 2.2.1, in presenza di nodi la distanza tra gli estremi del filamento in tensione è minore della sua lunghezza. In caso contrario ci si aspetta che i due valori tendano a coincidere. Si riportano i valori misurati della differenza tra questi due dati nel grafico in *Figura 2.8*. Ci si riferirà questo dato come δL .

Come si può osservare, la maggior parte delle proteine presenta un δL di quasi 4-5 Å. Questa è circa la distanza tra un aminoacido e il successivo e non è compatibile con lo sviluppo di un nodo. Si ritiene piuttosto che questo valore sia dovuto al fatto che le velocità dei due estremi al termine della simulazione non sono esattamente uguali. Lasciando trascorrere ulteriore tempo per smorzare tutte le oscillazioni del filamento sarebbe possibile completare l'allungamento. Si conclude che nelle simulazioni in questione non si sia sviluppato un nodo, salvo avvenuta compenetrazione.

Alcuni stiramenti hanno portato però alla rilevazione di δL ben' maggiori. Come già accennato, per studiare il fenomeno della compenetrazione si era sviluppato un toy-model nel quale si poteva osservare un nodo a trifoglio in tensione. Operando uno stiramento e misurando il δL anche in questo toy-model si è osservato che questo risulta 40 Å. In base a questa osservazione si sostiene che le simulazioni che presentano un valore di δL compatibile con questo valore siano da considerarsi annodate. È chiaro che non si può imporre l'uguaglianza esatta a questo valore per stabilire l'annodamento. Come range di compatibilità si è scelto un intervallo centrato su 40 Å. Per le simulazioni che non si considerano annodate si prevede un δL di 0 Å ma se ne misura uno di circa 4 Å. Si è scelto quindi questo valore come semiampiezza dell'intervallo di compatibilità. Si tenga presente che il dato 40 Å è specifico per il nodo a trifoglio. La presenza di altri nodi porterebbe sicuramente a un diverso valore di questo parametro. Per questa ragione definiamo il δL specifico del nodo a trifoglio come δLt . Dalla *Figura 2.8* possiamo osservare alcuni δL che corrispondono a queste aspettative. Poiché è possibile che si sviluppino più di un nodo nello stesso sistema, si considereranno doppiamente annodate le simulazioni con δL che ricada in un intervallo centrato in $2 * \delta Lt$, della stessa ampiezza del precedente.

Alcuni dati di δL però non ricadono in nessuno di questi intervalli. Altri in fine presentano un numero di nodi superiore a ogni legittima aspettativa. Per spiegare questi risultati vi sono due possibilità: la prima è che si sia sviluppato un nodo di diverso tipo da quello a trifoglio studiato sul toy-model, la seconda è che gli estremi del filamento abbiano raggiunto velocità uguali senza che lo stiramento fosse terminato. Quest'ultima

eventualità ha luogo quando l'attrito sviluppato in zone particolarmente ingarbugliate del filamento bilancia la tensione ai capi. Questa circostanza non costituisce nessun evidenza riguardo al fenomeno di annodamento. Ne impedisce però però la rilevazione ed è risultato impossibile ovviare a questa interferenza.

Visto il numero gestibile di simulazioni che presentano dati ambigui si è deciso di osservare gli stiramenti di persona. Tutte le proteine così esaminate sono risultate ingarbugliate tranne 1RA9. Questa proteina presenta un nodo del quale però non si è riusciti a fornire una classificazione.

Alle simulazioni prodotte è associata la probabilità che, durante il loro svolgimento, sia avvenuta una compenetrazione. Questa probabilità è ricavata dal confronto della durata della simulazione con la poissoniana con media $Tm = 83518$ s. Il valore di questo parametro dipende dalla scelta del parametro DT e dalla tensione applicata sugli estremi. Si è spiegato il motivo di questa scelta nello studio sulla compenetrazione della sezione 2.2.2. Questa probabilità costituisce una misura dell'incertezza sul dato ricavato dalla simulazione.

Si riportano le informazioni principali contenute nel database e i dati ricavati dagli stiramenti nella tabella 2.1.

| Nome | Aminoacidi | Folding | δL [Å] | Tempo [s] | Nodi | Incertezza |
|------|------------|-----------|----------------|-----------|------|------------|
| 1A6N | 151 | two-state | 5.792 | 1185 | 0 | <2.45 % |
| 1ARR | 53 | two-state | 4.0948 | 437 | 0 | <2.45 % |
| 1AYE | 201 | two-state | 4.872 | 1986 | 0 | <2.45 % |
| 1BDD | 30 | two-state | 4.0456 | 394 | 0 | <2.45 % |
| 1BNZ | 32 | two-state | 4.0712 | 373 | 0 | <2.45 % |
| 1C9O | 132 | two-state | 6.528 | 936 | 0 | <2.45 % |
| 1COA | 32 | two-state | 4.0292 | 417 | 0 | <2.45 % |
| 1CSP | 34 | two-state | 4.0224 | 447 | 0 | <2.45 % |
| 1ENH | 27 | two-state | 4.9884 | 242 | 0 | <2.45 % |
| 1G6P | 231 | two-state | 894.15 | 95 | ? | <2.45 % |
| 1HRC | 52 | two-state | 4.208 | 392 | 0 | <2.45 % |
| 1IMQ | 43 | two-state | 4.0924 | 387 | 0 | <2.45 % |
| 1L2Y | 380 | two-state | 487.864 | 4628 | ? | <2.45 % |
| 1L8W | 571 | two-state | 37.6 | 18593 | 1 | <8.33% |
| 1LOP | 84 | two-state | 46.7088 | 400 | ? | <2.45 % |
| 1NYF | 29 | two-state | 7.6644 | 96 | 0 | <2.45 % |
| 2VIK | 63 | two-state | 4.3752 | 450 | 0 | <2.45 % |
| 3CHY | 64 | two-state | 4.0552 | 487 | 0 | <2.45 % |
| 2AIT | 333 | two-state | 80.2088 | 1581 | 2 | <2.45 % |
| 2CI2 | 33 | two-state | 4.0784 | 392 | 0 | <2.45 % |
| 2CRO | 33 | two-state | 4.0108 | 458 | 0 | <2.45 % |
| 2LZM | 82 | two-state | 4.2524 | 570 | 0 | <2.45 % |
| 1URN | 144 | two-state | 4.036 | 1410 | 0 | <2.45 % |
| 1VII | 18 | two-state | 4.0548 | 348 | 0 | <2.45 % |
| 1VIK | 99 | two-state | 4.0628 | 770 | 0 | <2.45 % |
| 1YCC | 54 | two-state | 4.0396 | 479 | 0 | <2.45 % |
| 256B | 106 | two-state | 4.248 | 841 | 0 | <2.45 % |
| 2A5E | 78 | two-state | 4.0536 | 567 | 0 | <2.45 % |
| 1TEN | 45 | two-state | 4.1684 | 388 | 0 | <2.45 % |
| 1TIT | 45 | two-state | 4.1328 | 386 | 0 | <2.45 % |

| Nome | Aminoacidi | Folding | δL [Å] | Tempo [s] | Nodi | Incertezza |
|------|------------|------------|----------------|-----------|------|------------|
| 1SCE | 204 | two-state | 41.744 | 2263 | 1 | <2.45 % |
| 1PCA | 202 | two-state | 5.04 | 2011 | 0 | <2.45 % |
| 1PGB | 28 | two-state | 4.0128 | 450 | 0 | <2.45 % |
| 1PHP | 197 | two-state | 5.776 | 2044 | 0 | <2.45 % |
| 1PKS | 38 | two-state | 4.072 | 400 | 0 | <2.45 % |
| 1POH | 43 | two-state | 4.0492 | 431 | 0 | <2.45 % |
| 1PSF | 35 | two-state | 4.0468 | 415 | 0 | <2.45 % |
| 1RA9 | 80 | two-state | 64.34 | 780 | ?* | <2.45 % |
| 1PHP | 197 | multistate | 5.704 | 2043 | 0 | <2.45 % |
| 1PIN | 78 | multistate | 4.0744 | 617 | 0 | <2.45 % |
| 1RIS | 49 | multistate | 4.0532 | 421 | 0 | <2.45 % |
| 1SRL | 28 | multistate | 4.8396 | 238 | 0 | <2.45 % |
| 1UBQ | 38 | multistate | 4.0528 | 402 | 0 | <2.45 % |
| 2ACY | 49 | multistate | 4.1492 | 437 | 0 | <2.45 % |
| 2PDD | 744 | multistate | 5.66 | 22514 | 0 | <8.33 % |
| 2RN2 | 78 | multistate | 4.056 | 685 | 0 | <2.45 % |
| 3CHY | 64 | multistate | 4.0552 | 487 | 0 | <2.45 % |
| 1ARR | 53 | multistate | 4.038 | 460 | 0 | <2.45 % |
| 1AEY | 435 | multistate | 384.444 | 7024 | ? | <2.45 % |
| 1BRS | 294 | multistate | 172.68 | 4288 | ? | <2.45 % |
| 1BTA | 45 | multistate | 4.1132 | 379 | 0 | <2.45 % |
| 1CBI | 136 | multistate | 6.84 | 1192 | 0 | <2.45 % |
| 1CEI | 43 | multistate | 4.0368 | 440 | 0 | <2.45 % |
| 1EAL | 318 | multistate | 537.4 | 3076 | ? | <2.45 % |
| 1FNF | 184 | multistate | 5.36 | 1513 | 0 | <2.45 % |
| 1GXT | 45 | multistate | 4.1088 | 416 | 0 | <2.45 % |
| 1HEL | 65 | multistate | 4.0896 | 494 | 0 | <2.45 % |
| 1HNG | 175 | multistate | 4.848 | 1452 | 0 | <2.45 % |
| 1IFC | 131 | multistate | 5.62 | 941 | 0 | <2.45 % |
| 1JON | 71 | multistate | 4.0852 | 534 | 0 | <2.45 % |
| 1OPA | 133 | multistate | 78.56 | 1049 | 2 | <2.45 % |

Tabella 3.1: Nelle colonne troviamo, in ordine: nome della proteina, numero di aminoacidi da cui è composta, modalità di folding (two-state o multistate), differenza tra lunghezza totale del backbone e distanza tra gli estremi al termine dello stiramento, tempo richiesto per lo stiramento, numero di nodi stimato, incertezza del dato dovuta al fenomeno di compenetrazione. Nella colonna che contiene il numero di nodi si segnala con un punto interrogativo una configurazione non identificabile né con un nodo a trifoglio, né con il filamento completamente disteso e nemmeno con altri tipi di nodi. La presenza di un nodo non identificato ma diverso da quello a trifoglio è segnalata con "?*"

| | two-state | multistate | totale |
|---------|-----------|------------|--------|
| nodi si | 4 | 1 | 5 |
| nodi no | 31 | 19 | 50 |
| totale | 35 | 20 | 61 |

Tabella 3.2: Schema dei dati utilizzati per il test di Fisher

Da quanto si può dedurre da questi dati il fenomeno di annodamento ha un'occorrenza abbastanza bassa per entrambi i tipi di proteine. Solo il 9% di quelle studiate ne ha prodotti.

È stato eseguito un test esatto di Fisher sul campione valutando la correlazione tra la presenza (in qualsiasi numero) di nodi e la modalità di folding. Si riportano i dati analizzati con questo test in tabella 2.3. Segue il calcolo della probabilità di ottenere un campione come quello studiato da due osservabili non correlati (P_n). Si sono esclusi dal calcolo i dati sulle proteine con configurazione ambigua ma si è conservato il nodo su 1RA9.

$$P_n = \frac{5! * 50! * 35! * 20!}{4! * 1! * 31! * 19! * 55!} = 0,3$$

Il test di Fisher mette in evidenza una probabile correlazione tra la presenza di nodi e il folding two-state. È bene tener conto che comunque le configurazioni ambigue, escluse dal calcolo, potrebbero spostare di molto questa statistica qualora si trovasse un modo di risolverle. Queste infatti sono in numero maggiore ai nodi rilevati. Non si è in grado di fornire nessun ipotesi per spiegare un eventuale correlazione tra i due osservabili.

Come si è detto nella sezione 1.2.1 la procedura utilizzata per lo stiramento di tutte le proteine non è l'unica possibile. Procedure diverse possono dare luogo a risultati differenti e dunque ha senso approfondire la questione. Si è voluto provare altre procedure particolare variando la direzione della tensione applicata sui capi. Per motivi logistici sono state oggetto di questa prova solo le proteine più corte di 200 aminoacidi. Si è ripetuto lo stiramento nella direzione del primo set di simulazioni e come direzioni alternative ne sono state scelte due ortogonali alla prima e anche tra loro. Otteniamo dunque un set di tre direzioni ortogonali di cui una è la congiungente tra gli estremi. Si sono testate procedure che applicassero una tensione parallela a i nuovi assi, in tutti e due i sensi. Sono stati svolti stiramenti anche sull'asse iniziale in senso opposto. Si ottengono in totale sei stiramenti per ogni proteina di meno di 200 aminoacidi. Si sono anche confrontate le misure di δL , oltre che con il δL inerente al nodo a trifoglio, anche con quello rilevato sulla proteina 1RA9. Il nodo su di essa non è stato identificato, ciò non di meno lo si considera un nodo. Si associa a questo nodo un intervallo di δL centrato nel valore $\delta L^* = 64 \text{ \AA}$ della stessa ampiezza di quelli relativi al nodo a trifoglio. Si riporta il risultato di questi stiramenti in tabella 2.2.

| Nome | Aminoacidi | nodi A | nodi B | nodi C | nodi -A | nodi -B | nodi -C |
|------|------------|--------|--------|--------|---------|---------|---------|
| 1A6N | 151 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1ARR | 53 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1BDD | 30 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1BNZ | 32 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1BTA | 45 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1C9O | 132 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1CBI | 136 | 0 | ? | ? | ? | 2 | 1 |
| 1CEI | 43 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1COA | 32 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1CSP | 34 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1ENH | 27 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1FNF | 184 | 0 | 1 | 0 | 0 | ?* | 0 |
| 1GXT | 45 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1HEL | 65 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1HNG | 175 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1HRC | 52 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1IFC | 131 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1IMQ | 43 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1JON | 71 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1LOP | 84 | 1 | 0 | ? | ? | 0 | 0 |
| 1NYF | 29 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1OPA | 133 | 2 | ? | 2 | ? | 1+?* | 2 |

| Nome | Aminoacidi | nodi A | nodi B | nodi C | nodi -A | nodi -B | nodi -C |
|------|------------|--------|--------|--------|---------|---------|---------|
| 1PGB | 28 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1PHP | 197 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1PHP | 197 | 0 | 0 | 0 | 0 | 0 | ? |
| 1PIN | 78 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1PKS | 38 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1POH | 43 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1PSF | 35 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1RA9 | 80 | ?* | 0 | 0 | 0 | 1 | 0 |
| 1RIS | 49 | 0 | ? | 0 | 0 | 0 | 0 |
| 1SRL | 28 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1TEN | 45 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1TIT | 45 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1UBQ | 38 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1URN | 144 | 0 | 0 | ? | 0 | 0 | 0 |
| 1VII | 18 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1VIK | 99 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1YCC | 54 | 0 | 0 | 0 | 0 | 0 | 0 |
| 256B | 106 | 0 | ? | 0 | ? | 0 | 0 |
| 2A5E | 78 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2ACY | 49 | 0 | 0 | 0 | 0 | 1 | 0 |
| 2CI2 | 33 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2CRO | 33 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2LZM | 82 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2RN2 | 78 | 0 | 0 | 1 | 0 | 0 | 0 |
| 2VIK | 63 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3CHY | 64 | 0 | 0 | 0 | 0 | 0 | 0 |

Tabella 3.3: In tabella sono riportati nome della proteina, numero di aminoacidi che la compongono e l'occorrenza di nodi al termine dello stiramento secondo la procedura a cui si fa riferimento. Ci si riferisce alla procedura attraverso l'asse lungo il quale si è esercitata la tensione. L'asse A è l'asse che congiunge gli estremi e gli assi B e C sono assi a e tra loro lui ortogonali. Gli stiramenti -A, -B e -C si è applicata una tensione lungo gli assi omonimi, ma nel verso opposto. Il risultato ambiguo di uno stiramento è rappresentato da un punto interrogativo e quelle riconducibili al nodo non identificato rilevato su 1RA9 con ?*

Dai dati riportati possiamo notare innanzitutto che il fenomeno di annodamento, come previsto, dipende strettamente dalla direzione della tensione e in generale dalla procedura utilizzata per ricavare il dato della distanza tra gli estremi. Alcune proteine mostrano in ogni stiramento assenza di nodi ma nessuna di quelle che presenta un nodo lo conserva in tutti gli stiramenti. Sono stati rilevati un gran numero di dati ambigui. Molti di questi sono stati prodotti da proteine che, in stiramenti diretti su diverse angolazioni, hanno dato luogo a nodi. Questo fa sospettare che alcune proteine abbiano una struttura meno predisposta allo stiramento e in qualche modo più ingarbugliata. Non si è in grado però di produrre evidenze sperimentali al riguardo.

Si tenga anche conto infine che nel rappresentare le proteine si sono semplificati molti aspetti della loro struttura: le interazioni esterne al backbone e la diversa massa degli aminoacidi prima di tutto. Non si è potuto dunque nemmeno avere la certezza che, ripetendo in laboratorio su una proteina una procedura analoga a quella simulata, questa dia luogo al risultato osservato in questo studio.

Capitolo 4

Conclusioni

Gli obiettivi che ci si era posti in questo studio erano i seguenti:

- Sviluppare un software in grado di simulare la meccanica di un filamento e testarne potenzialità e limiti.
- Definire il fenomeno di annodamento su varietà unidimensionali con estremi e riprodurlo nella simulazione sviluppata.
- Testare la simulazione su dati reali, in questo caso la struttura delle proteine, e studiare l'occorrenza dell'annodamento.

Si può dire di essere riusciti a sviluppare una simulazione che sia in grado di riprodurre in modo soddisfacente un filamento, pur facendo i conti con limiti di varia natura.

È stato possibile formulare una definizione di nodo su varietà con estremi non univoca, ma comunque autoconsistente.

La simulazione è in grado di riprodurre nodi di varia natura e ha permesso di studiare l'occorrenza di questo fenomeno in buona parte del database di proteine a disposizione (86%). Non si è riusciti però a studiare proteine troppo lunghe o troppo ingarbugliate. Da questo studio emerge che il fenomeno di annodamento è abbastanza raro nelle proteine. Seguendo la procedura indicata nella sezione 1.2.1 solo il 9% dei campioni studiati ha prodotto annodamenti. Solo uno su cinque di questi annodamenti è risultato diverso dal nodo a trifoglio. Questo dato induce a ritenere questo nodo in qualche modo più probabile di tutti gli altri. Dal confronto dell'occorrenza di annodamenti con il tipo di folding emerge la possibilità che le proteine two-state siano più soggette a sviluppare nodi rispetto a quelle multistate. Si è eseguito un test di Fisher sugli osservabili two-state/multistate e nodi sì/nodi no. Risulta che solo nel 30 % dei casi due osservabili non collegati produrrebbero dati simili a quelli riportati. Va però tenuto in considerazione il fatto che le proteine ingarbugliate sono in numero maggiore rispetto a quelle che sono risultate essere annodate. Riuscire a studiare anche queste proteine potrebbe cambiare di molto il risultato del test di Fisher.

Si tenga anche presente che nella simulazione si sono operate semplificazioni drastiche della struttura della proteina. Gli aminoacidi sono stati considerati tutti di massa uguale e non si è tenuto conto delle loro interazioni chimiche tra di loro e con l'ambiente esterno. Non è possibile dunque utilizzare questa simulazione come modello predittivo. L'esito di una procedura simile a quella descritta svolta in laboratorio potrebbe essere assai diverso.

4.1 Possibilità di sviluppo

Testare la simulazione su dati reali ha permesso di evidenziare i limiti più pesanti del programma e della procedura per rilevare i nodi.

Il primo è costituito dalla lentezza della simulazione per sistemi che presentano molti punti. Lo stiramento delle proteine più lunghe ha richiesto ore, e quelle con più di 800 aminoacidi non sono state nemmeno prese in considerazione perché la macchina in uso non risultava in grado nemmeno di dare inizio alla simulazione. Questo problema è dovuto principalmente a due fatti. In primo luogo siamo costretti a usare DT molto ridotti per scongiurare la compenetrazione e dunque serve un grande numero di iterazioni per completare uno stiramento. Inoltre il tempo di ogni iterazione cresce quadraticamente con il numero di aminoacidi.

Soluzione a entrambi questi problemi potrebbe essere quella di trattare i punti come campi di forza piuttosto che come sfere rigide. In questo modo si potrebbe evitare di utilizzare una funzione ad hoc per lo sviluppo degli urti (*SviluppaUrti*) e lasciare che anche quelli siano risolti da RK4. Questo potrebbe rendere il sistema meno vulnerabile alla compenetrazione e dunque permettere di adottare DT più ampi per far evolvere più velocemente il sistema. Si ritiene infatti che, come si è argomentato nel paragrafo 2.2.2, la compenetrazione sia dovuta proprio alla divisione delle funzioni per la gestione degli urti da RK4. Si potrebbe inoltre implementare più comodamente anche le interazioni tra punti non attigui per rappresentare legami esterni al backbone.

Un altro limite emerge quando si esamina un filamento particolarmente ingarbugliato e l'attrito blocca lo stiramento. Aumentare la tensione ai capi non risolverebbe necessariamente il problema in tutte le configurazioni ed esporrebbe il sistema alla compenetrazione. Inoltre si è stati in grado di identificare la classe di appartenenza del nodo solo grazie al fatto che si era sviluppato un toy-model con un nodo a trifoglio. È stato rilevato anche un nodo di una classe differente, ma non si è stati in grado di identificare quale classe fosse. Questi limiti sono inerenti alla procedura scelta, la quale implica operazioni fisiche come lo stiramento e non ha altro modo per distinguere i nodi se non il dato δL . Dunque per risolvere questi problemi si potrebbe definire un'altra procedura. Si consideri la seguente variazione di quella utilizzata per questo studio ed esposta al paragrafo 1.2.1

- Partendo dalla condizione iniziale sarà sempre necessario portare gli estremi in una condizione periferica del sistema. È possibile fare ciò attraverso operazioni meccaniche (come fatto in questo studio), ma anche attraverso prolungamenti della varietà.
- Fatto ciò si dovrà proiettare il nostro filamento su un piano e congiungerne gli estremi senza dare luogo a ulteriori incroci.
- A questo punto la nostra proiezione sarà conforme a quelle di varietà senza estremi. Si possono dunque applicare gli algoritmi formulati riguardo ai nodi su varietà chiuse. In questo modo si potrebbe ricavare gli invarianti del nodo e ottenere un parametro per identificare la sua classe.

Questa procedura ha il vantaggio evidente che non necessita quasi assolutamente di far evolvere il sistema. Un programma che si basasse su di essa non avrebbe bisogno di nessuna funzione simile a RK4 per integrare le equazioni del moto. Rimane il problema di come implementare gli algoritmi della teoria dei nodi. Si è fatto un tentativo in questa direzione cercando di implementare l'algoritmo per il calcolo del polinomio di Jones, ma ulteriori sviluppi sono necessari per poter trarre conclusioni, anche preliminari, sull'efficacia dell'approccio.

Capitolo 5

Ringraziamenti

Si ringrazia il Professor Sandro Rambaldi (dipartimento di fisica e astronomia di Bologna) per aver sviluppato la base di questo software utilizzato per produrre la simulazione.

Bibliografia

- [1] Mary K. Campbel, Shawn O. Farrel, *Biochemistry 7th edition*, 2012, pubblicato da Brooks/Cole, Cengage Learning
- [2] Colin C. Adams, *The knot book*, 1994, pubblicato da W.H. Freeman and Company
- [3] Louis H. Kauffman *Knot and Physics*, 1991 pubblicato da Wuorld Scientific publishing
- [4] Nature New Biol, *Protein Data Bank*, Web, <https://www.rcsb.org>
- [5] Graziano Servizi, *C++ Commedia*, Web, http://www.physycom.unibo.it/labinfo/per_cominciare.php
- [6] Menichetti Giulia, Piero Fariselli e Daniel Remondini, *Network measures for protein folding state discrimination*. Scientific Reports 6 (2016): pagina 30367.