

## **SCHEDA LABORATORIO SCIENTIFICO n. 14**

**Chimica Farmaceutica Computazionale – Computational Medicinal Chemistry**

**Settore Scientifico-Disciplinare di riferimento:** CHIM/08

**Responsabile:** Sandro Cosconati

**RADoR:** Sandro Cosconati

**Tipologia:** COMPUTAZIONALE

**Gruppi afferenti:** Drug Discovery: Progettazione, Sintesi e Veicolazione (DSD Lab)

### **LOCALIZZAZIONE E DESCRIZIONE**

- Primo piano del corpo A del DISTABIF, (locale 3°13.11);
- Dimensioni: 13,7 m<sup>2</sup>
- n. 6 postazioni di lavoro

### **ATTIVITÀ SVOLTE NEL LABORATORIO**

- Docking molecolare
- Dinamica Molecolare
- Programmazione in Python

### **RELAZIONE SINTETICA DESCRITTIVA DELLE ATTIVITÀ SVOLTE E DELLE MODALITÀ OPERATIVE**

#### **1. Docking molecolare**

Preparazione della proteina bersaglio: La struttura tridimensionale della proteina bersaglio viene solitamente ottenuta tramite tecniche sperimentali come la cristallografia a raggi X o la risonanza magnetica nucleare (NMR). La proteina viene quindi preparata per il docking, rimuovendo eventuali molecole co-cristallizzate, aggiustando la geometria e assegnando cariche e idrogeni.

Preparazione del ligando: Il ligando (ad esempio un candidato farmaco) viene anch'esso preparato, spesso partendo dalla sua struttura tridimensionale ottenuta tramite tecniche sperimentali o previsioni computazionali. Anche il ligando viene preparato assegnando cariche, idrogeni e ottimizzando la sua geometria.

Generazione dei conformeri: Sia il ligando che la proteina possono esistere in diverse conformazioni. In questa fase, vengono generate diverse conformazioni del ligando e della proteina per esplorare una vasta gamma di possibili orientamenti e interazioni.

Docking: Utilizzando algoritmi di docking, il software cerca di trovare l'orientamento ottimale del ligando all'interno del sito attivo della proteina bersaglio. Questo viene fatto valutando l'energia di legame tra il ligando e la proteina in diverse pose (configurazioni).

Scoring e valutazione: Le diverse pose generate durante il docking vengono valutate e classificate in base a un punteggio di affinità, che tiene conto di fattori come l'energia di legame, la complementarità dei contorni e le interazioni intermolecolari. Le pose con i punteggi più alti sono considerate le più probabili e promettenti.

Analisi dei risultati: Le pose migliori vengono quindi analizzate per identificare i dettagli delle interazioni tra il ligando e la proteina bersaglio, al fine di comprendere meglio i meccanismi di legame e guidare lo sviluppo di nuovi composti con maggiore affinità e selettività.

#### **2. Dinamica Molecolare**

Preparazione del sistema: Inizialmente, è necessario preparare il sistema di interesse, che può comprendere una o più molecole (come proteine, ligandi, solventi, ioni, ecc.). Questo può includere la definizione della struttura tridimensionale delle molecole, l'assegnazione delle cariche e delle masse agli atomi, e la definizione delle condizioni iniziali (posizioni, velocità, temperatura, pressione, ecc.).

Scelta del potenziale energetico: Per simulare il comportamento delle molecole, è necessario utilizzare un modello di potenziale energetico che descriva le interazioni tra gli atomi e le molecole nel sistema. Questo può includere termini

per legami chimici, angoli di legame, torsioni, interazioni a lungo raggio (ad esempio, interazioni elettrostatiche) e interazioni a corto raggio (ad esempio, forze di van der Waals).

**Integrazione delle equazioni del moto:** Una volta definito il potenziale energetico, le equazioni del moto per tutte le particelle nel sistema vengono risolte numericamente per determinare le loro posizioni e velocità nel tempo. Questo viene fatto utilizzando metodi numerici come l'integrazione di Verlet, l'algoritmo di leapfrog o altri algoritmi di dinamica molecolare.

**Simulazione temporale:** La simulazione viene eseguita per un certo periodo di tempo, durante il quale le posizioni e le velocità delle particelle nel sistema vengono aggiornate in base alle forze che agiscono su di esse. Durante la simulazione, vengono registrate varie proprietà del sistema, come l'energia totale, la temperatura, la pressione, e le traiettorie delle particelle.

**Analisi dei risultati:** Una volta completata la simulazione, i dati raccolti possono essere analizzati per comprendere il comportamento dinamico del sistema. Questo può includere l'analisi delle traiettorie delle particelle, il calcolo di grandezze termodinamiche come l'energia libera, l'entropia e la capacità termica, e lo studio delle interazioni molecolari e dei cambiamenti conformazionali nel sistema.

### **3. Programmazione in Python**

**Pianificazione e Analisi del Problema:** Prima di iniziare a scrivere il codice, è importante comprendere appieno il problema che si intende risolvere. Questo può includere la definizione dei requisiti, l'analisi delle possibili soluzioni e la pianificazione dell'approccio da seguire.

**Scrittura del Codice:** Una volta compreso il problema, si passa alla scrittura del codice Python per implementare la soluzione. Questo può includere la definizione di variabili, la scrittura di istruzioni condizionali, cicli, funzioni e altro ancora, a seconda delle necessità del problema.

**Testing e Debugging:** Dopo aver scritto il codice, è importante testarlo per assicurarsi che funzioni come previsto. Questo può includere l'esecuzione di test automatizzati, l'input manuale di dati di test e l'analisi dei risultati per individuare eventuali errori o bug nel codice.

**Ottimizzazione e Refactoring:** Una volta che il codice funziona correttamente, è possibile ottimizzarlo per migliorare le prestazioni o la leggibilità. Questo può includere la semplificazione del codice, l'eliminazione di duplicati, l'ottimizzazione degli algoritmi e altro ancora.

**Documentazione:** È importante documentare il codice in modo chiaro e conciso, in modo che altri programmatori possano comprendere facilmente il suo funzionamento. Questo può includere commenti nel codice, documentazione esterna e guide per gli utenti.

**Gestione del Codice Sorgente:** Durante lo sviluppo di progetti più complessi, è utile utilizzare strumenti di gestione del codice sorgente come Git per tenere traccia delle modifiche al codice e collaborare con altri sviluppatori.

**Deployment e Manutenzione:** Una volta completato e testato il codice, è possibile distribuirlo in produzione. Questo può includere la configurazione di un ambiente di produzione, l'installazione del codice e la gestione delle eventuali manutenzioni o aggiornamenti futuri.

### **LISTA DELLE ATTREZZATURE PRESENTI:**

- 6 Workstations multicore.

### **LISTA DEI DISPOSITIVI DI PROTEZIONE GENERALE (DPG)**

- Nessuno

### **LISTA DEI DISPOSITIVI DI PROTEZIONE INDIVIDUALI (DPI) AD USO PERSONALE DEGLI OPERATORI.:**

- Nessuno

### **Categorie ISI WEB di riferimento:**

Chemistry, Medicinal

### **Categorie ERC di riferimento:**

#### **❖ PE5 Synthetic Chemistry and Materials**

- ❖ PE5\_18 Intelligent materials synthesis – self assembled materials