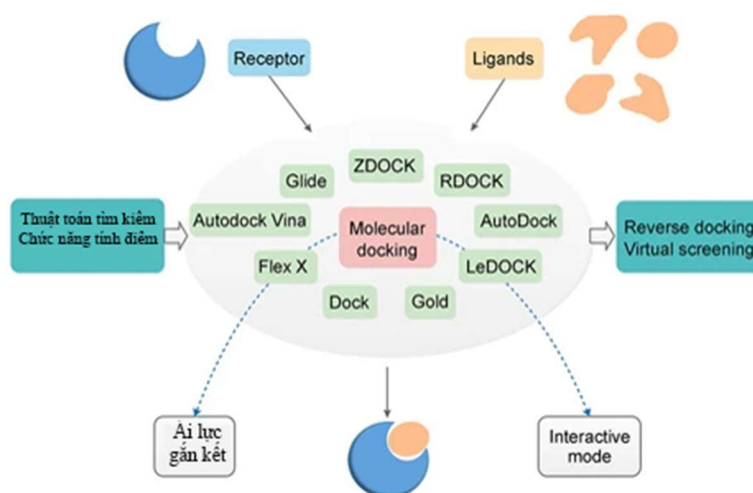


1 MÔ HÌNH DOCKING PHÂN TỬ

1.1 Khái niệm docking

Mô hình gắn kết phân tử (Molecular Docking) là một phương pháp có vai trò lớn trong thiết kế thuốc với sự trợ giúp của máy tính (CADD – Computer-Aided Drug Design). Phương pháp này được dùng để đánh giá và dự đoán khả năng gắn kết của phân tử nhỏ (ligand) vào cấu trúc mục tiêu (receptor). Hiện nay, phương pháp docking đang được sử dụng phổ biến vì tính đơn giản, dễ sử dụng và ít tốn thời gian trong các phương pháp sàng lọc ảo. Ứng dụng của docking bao gồm nghiên cứu sự liên quan giữa cấu trúc–tác dụng, tối ưu hoá chất khởi nguồn (lead optimization), tìm các chất khởi nguồn tiềm năng (lead compound), nghiên cứu các đột biến dựa trên việc đưa ra tiên đoán về giả thuyết gắn kết, hỗ trợ đưa các chất nền hoặc chất ức chế vào bản đồ mật độ electron (electron density) trong các cấu trúc nhiễu xạ tia X,...¹¹¹



Hình 1.1. Mô hình docking phân tử¹¹²

Quá trình docking gồm 3 giai đoạn chính:

- Giai đoạn 1: tạo cấu dạng gắn kết. Dựa trên sự tính toán năng lượng tĩnh điện và lực *Van der Waals*, cấu dạng thích hợp nhất cho khả năng gắn kết vào protein của các hợp chất sẽ được xác định. Giai đoạn này thường tạo ra nhiều cấu dạng khác

nhau. Tiếp theo các cấu dạng gắn kết được chấm điểm dựa trên mức độ tương thích giữa cấu dạng đó với vùng gắn kết trên đích tác động.

- Giai đoạn 2: chấm điểm, có thể giúp ước đoán năng lượng gắn kết.
- Giai đoạn 3: xếp hạng. Tính toán năng lượng tự do của sự gắn kết một cách chính xác nhất có thể, dựa trên các tính toán phức tạp liên quan đến năng lượng entropy và solvat hóa.

Docking gắn kết phối tử vào cấu trúc mục tiêu theo ba phương thức sau:

- Docking cứng: phối tử và protein mục tiêu đều cố định
- Docking bán linh động: phối tử là phần tử linh động, protein mục tiêu cố định
- Docking linh động hoàn toàn: phối tử và protein mục tiêu đều linh động

1.2 Quá trình gắn kết phân tử

Quá trình mô phỏng sự gắn kết giữa phối tử và protein được thực hiện qua hai bước cơ bản là lựa chọn cấu dạng phù hợp của phối tử (sampling) và đánh giá kết quả phức hợp được tạo thành (scoring).^{113,114}

- Chọn cấu dạng phối tử tương thích với khoang gắn kết có thể được thực hiện bởi nhiều thuật toán khác nhau, được chia thành ba nhóm:
 - Tìm kiếm hệ thống (systematic search) bao gồm thuật toán dựa trên các phân mảnh (fragment-based), thuật toán xây dựng tăng cường (incremental construction algorithms),...
 - Tìm kiếm ngẫu nhiên (stochastic search) bao gồm giải thuật di truyền (genetic algorithm), thuật toán Monte carlo, thuật toán tìm kiếm Tabu,...
 - Tìm kiếm quyết định hay mô hình hoá (deterministic or simulation search) bao gồm mô phỏng động lực học phân tử (molecular dynamics simulation), thuật toán ủ mô phỏng SA (simulated annealing).¹¹⁴

- Hàm chấm điểm được chia thành bốn loại chính là chấm điểm dựa trên vật lý (physics-based), dựa trên kinh nghiệm (empirical), dựa trên hiểu biết (knowledge-based) và dựa trên học máy (machine learning-based).¹¹⁴

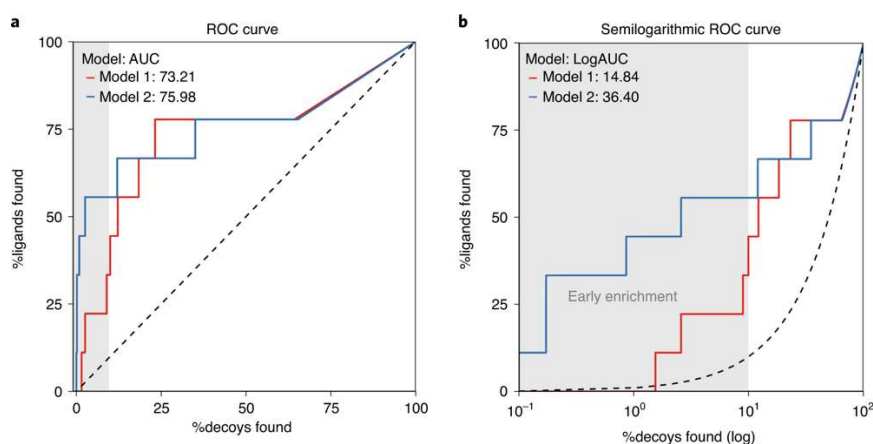
1.3 Đánh giá mô hình docking phân tử

Sự gắn kết các cấu dạng của ligand với protein và điểm số docking của các kiểu gắn kết vẫn cho kết quả gần đúng cao. Không giống như các phương pháp như cơ học lượng tử hoặc các phép tính mạng tinh thể nhất định trong cơ học thống kê, docking đã từ bỏ “ground truth” để có thể tìm kiếm một cách thực tế giữa một không gian hóa học rộng lớn và đang phát triển. Do đó, các phép tính kiểm tra hồi chứng là rất quan trọng đối với sự thành công của một mô hình docking. Với các chất đối chứng, chúng không đảm bảo mô hình chắc chắn đúng, nhưng chúng giúp xác định nguy cơ thất bại rõ ràng và có thể giúp người ta hiểu vấn đề sai ở đâu nếu chúng xảy ra. Thông qua một phép kiểm soát, đánh giá xem liệu các tham số gắn kết đã chuẩn bị có thể ưu tiên các phối tử có hoạt tính hơn các phân tử không hoạt tính hay không. Trong một túi gắn kết được tối ưu hóa, các chất hoạt tính đã biết trước sẽ xếp hạng cao hơn so với đường nền của các phân tử môi nhử trong thử nghiệm hồi chứng và các cấu dạng hợp lý phải được dự đoán. Nói một cách ngắn gọn, thử nghiệm hồi chứng giúp giảm thiểu các sai số loại 2 trong sàng lọc ảo (dương tính giả), nhằm kiểm tra hiệu năng của mô hình docking về năng lực gắn kết (docking power) và năng lực sàng lọc (screening power) cao hay thấp bao gồm re-docking để kiểm tra năng lực gắn kết của phần mềm; và cross-docking cho kiểm tra năng lực sàng lọc của mô hình.

Việc kiểm tra hồi chứng được thực hiện bằng cách sử dụng những chất có hoạt tính đã biết trước và những chất không có hoạt tính (hoặc môi nhử) được tạo ra từ những chất có hoạt tính để mô phỏng gắn kết vào cấu trúc mục tiêu, kết quả thu được sẽ gợi ý ngưỡng điểm docking phù hợp nhất cho các chất gắn kết tiềm năng. Thông thường tập hoạt tính nên bao gồm 10 – 30 chất có cấu trúc đa dạng là tối ưu cho việc thực hiện kiểm tra hồi chứng.¹¹⁵ Mô hình docking được đánh giá dựa trên đại lượng đường cong ROC, diện tích dưới đường cong (AUC), logAUC và hệ số làm giàu trong thử

nghiệm hồi chứng. Ngoài ra, đại lượng trung bình hình học (G-Mean) cũng được sử dụng nhằm xác định điểm gãy (cut-off) cho đường cong ROC. G-mean là một đại lượng đo lường sự cân bằng giữa hiệu năng phân loại trên cả lớp đa số và thiểu số. G-mean thấp là một dấu hiệu của hiệu năng kém trong việc phân loại các chất có hoạt tính ngay cả khi các trường hợp không hoạt tính được phân loại rất chính xác.¹¹⁶

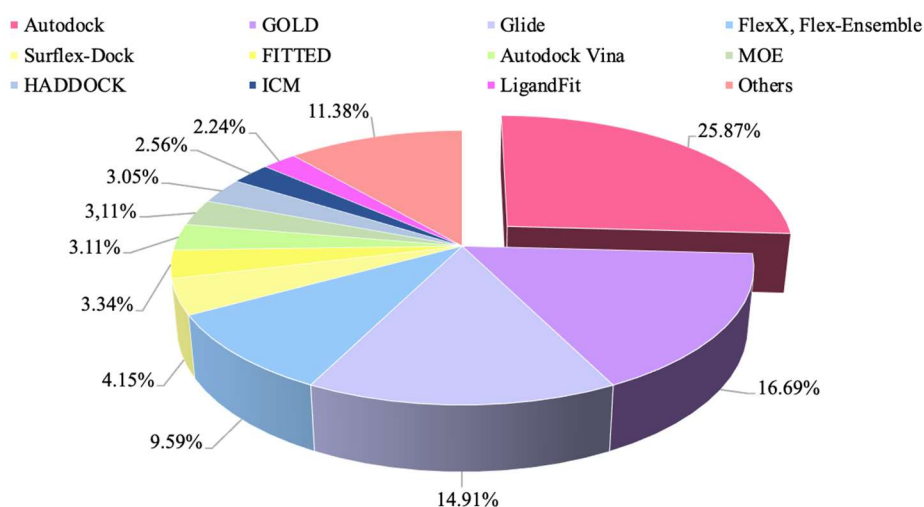
$$\text{G-mean} = \sqrt{\text{Độ nhạy} \times \text{Độ đặc hiệu}}$$



Hình 1.2. Đánh giá thử nghiệm hồi chứng mô hình docking¹¹⁵

1.4 Các phần mềm Docking

Trong đó, phương pháp docking bán linh động là phương pháp phổ biến nhất. Có nhiều phần mềm có thể docking được mô tả trong Hình 1.25.



Hình 1.3. Số lượng trích dẫn các bài báo khoa học của một số phần mềm gắn kết phổ biến nhất, được phân tích từ ISI Web of Science (2005) ¹¹⁷

Từ mã nguồn mở của Autodock Vina, các nhóm nghiên cứu trên khắp thế giới đã tiến hành cải tiến và sửa đổi, điển hình như thư viện QuickVina2 đã cải tiến thuật toán tìm kiếm, thư viện Smina cho phép người dùng tùy biến các tham số của hàm tính điểm, và cải thiện hàm tính điểm cho gắn kết với carbohydrat (Vina-Carb), liên kết với halogen (VinaXB),¹¹⁸ cũng như xếp hạng và cho điểm (Vinardo).

1.4.1 Các phần mềm dựa trên hàm chấm điểm kinh nghiệm

Từ mã nguồn mở của phần mềm Autodock Vina 1.1.2, nhiều phần mềm Docking đã được phát triển với những đặc điểm nổi bật như:

- **Smina:** có thêm nhiều hỗ trợ định dạng cho phối tử như *.pdb, *.sdf, thêm các định nghĩa mới trong hàm tính điểm (desolvation, electrostatics,...) và có thể tự điều chỉnh hàm tính điểm (đánh trọng số cho những hàm quan trọng).¹¹⁹
- **QVina2:** với tính năng mới là *first-order-consistency-check* nhằm cải thiện “significance test”, cải thiện tốc độ của quá trình tìm kiếm cục bộ. Kết quả là với tham số exhaustiveness bằng 8, Qvina2 cho tốc độ nhanh gấp 20,49 lần so với Vina 1.1.2, với hệ số tương quan Pearson giữa năng lượng gắn kết của Vina và Qvina 2 cho cấu dạng bền nhất là 0,967; và cho tất cả cấu dạng dự đoán là 0,911. Tuy

nhiên, Qvina2 cho kết quả tối đa là 20 cấu dạng dù có thay đổi tham số `num_modes`.¹²⁰

- **Autodock Vina 1.2.3:** với đặc điểm nổi bật là tích hợp được các tính năng sẵn có của Autodock 4¹²¹ như docking với kim loại, docking với macrocycle,... Tích hợp vào ngôn ngữ lập trình Python, giúp cho việc thiết kế phân tử thuốc trở nên dễ dàng hơn. Phát triển được batch docking, cho phép docking đồng thời nhiều phối tử. Cần lưu ý, đây là batch docking, có nghĩa là việc chuyển bị các thông số docking chỉ diễn ra 1 lần cho toàn bộ phối tử, khác với việc xây dựng một vòng lặp cho docking nhiều phối tử
- **Vina-GPU:** Vina-GPU¹²² là mã nguồn mở được xây dựng dựa trên Autodock Vina 1.1.2, với thuật toán tìm kiếm là Monte Carlo và hàm chấm điểm dựa trên kinh nghiệm. Điểm khác biệt là ở thuật toán tìm kiếm, dựa trên việc triển khai bằng OpenCL (GPU), giúp gia tăng đáng kể tốc độ tìm kiếm.¹²²

Các phần mềm này đều cùng sử dụng hàm chấm điểm dựa trên kinh nghiệm và sử dụng thuật toán Monte Carlo cho việc tối ưu hoá.

1.4.2 Phần mềm dựa trên hàm vật lý (Autodock-GPU)

Autodock 4 là phần mềm docking được sử dụng rộng rãi với hàm tính điểm dựa trên vật lý và đã được chứng minh hữu ích cho hàng loạt các công trình khám phá thuốc. Tuy nhiên, so với các phần mềm hiện đại gần đây, Autodock 4 cho thấy thời gian thực hiện lâu và khả năng ứng dụng hạn chế với quy mô lớn. Chính vì thế, Autodock-GPU,¹²³ với sự triển khai OpenCL trên Autodock 4, thúc đẩy kiến trúc song song của GPU, giảm thiểu thời gian docking lên tới 350 lần đối với quá trình đơn luồng. Hơn thế nữa, thuật toán tìm kiếm cục bộ ADADELTA, là phiên bản cải tiến của trình tối ưu hóa ngẫu nhiên Solis-Wets từ Autodock 4. Các thuật toán tìm kiếm cục bộ hiệu quả này làm giảm đáng kể số lượng lệnh gọi đến hàm tính điểm cần thiết để tạo ra kết quả tốt. Các cải tiến của Autodock-GPU, cả về thông lượng gắn kết và hiệu quả tìm kiếm, tạo điều kiện thuận lợi cho việc sử dụng chức năng chấm điểm Autodock 4 trong sàng lọc ảo quy mô lớn.

Hàm tính điểm của Autodock-GPU tương tự với Autodock 4.2.6 dựa trên trường lực cổ điển đánh giá năng lượng gắn kết bằng cách tính tổng của các tĩnh điện và *Van der Waals*. Các tương tác tĩnh điện được tính theo công thức Coulomb, còn các tương tác *Van der Waals* được mô tả bởi hàm Lennard-Jones.

$$v = W_{vdw} \sum_{i,j} \left(\frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^6} \right) + W_{hbond} \sum_{i,j} E(t) \left(\frac{C_{ij}}{r_{ij}^{12}} - \frac{D_{ij}}{r_{ij}^{10}} \right) + W_{elec} \sum_{i,j} \frac{q_i q_j}{\epsilon(r_{ij}) r_{ij}} + W_{sol} \sum_{i,j} (s_i v_j + s_j v_i) e^{(-r_{ij}^2/2\sigma^2)}$$

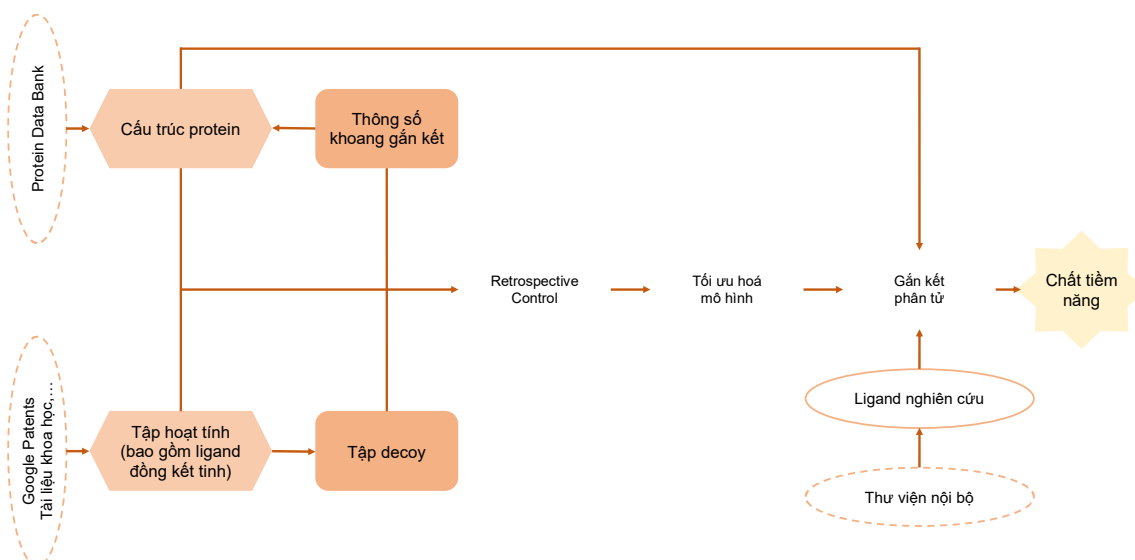
Đối với hai nguyên tử i, j , năng lượng nguyên tử cặp đôi được đánh giá bằng tổng tương tác *Van der Waals*, liên kết hydro, năng lượng coulomb và sự solvat hóa. W là trọng số để hiệu chỉnh năng lượng tự do theo kinh nghiệm.

1.4.3 So sánh các phần mềm docking

Bảng 1.1. So sánh các phần mềm docking

| Phần mềm docking | Hàm chấm điểm | Phương pháp tìm kiếm | Ưu điểm | Nhược điểm |
|----------------------------|---------------|---|---|--|
| Smina | Kinh nghiệm | Monte carlo – BFGS | Điều chỉnh được hàm tính điểm Xử lý đa luồng | Không có docking với kim loại |
| Qvina2 | Kinh nghiệm | Monte carlo – BFGS. Sử dụng <i>first-order-consistency-check</i> tăng tốc độ tìm kiếm | Tốc độ nhanh Xử lý đa luồng | Không có docking với kim loại |
| Autodock Vina 1.2.3 | Kinh nghiệm | Monte carlo – BFGS | Batch docking (dock nhiều ligand) Tích hợp python, xử lý đa luồng Có hàm tính điểm và các tính năng của AD4 | Không có docking với kim loại nếu dùng hàm kinh nghiệm |
| Vina-GPU | Kinh nghiệm | Monte carlo – BFGS | Sử dụng GPU - Tốc độ nhanh | Không có docking với kim loại |
| Autodock-GPU | Vật lý | ADADELTA | Sử dụng GPU - Tốc độ rất nhanh Batch docking | Khó cài đặt |

2 QUY TRÌNH



Hình 2.4. Quá trình xây dựng mô hình docking phân tử

2.1 Chuẩn bị cấu trúc protein và ligand đồng kết tinh

Cấu trúc của **Protein** được lấy từ ngân hàng dữ liệu protein trên Protein Data Bank (PDB ID:...) và được lưu ở định dạng *.pdb. Sử dụng Pymol để mở cấu trúc vừa tải về, kiểm tra cấu trúc của phức hợp protein-ligand, kiểm tra và xử những acid amin bị thiếu (missing residues) và những acid amin không đầy đủ bằng công cụ Protein builder trước khi tiến hành loại bỏ các phân tử dung môi, sau đó tiến hành tối thiểu hóa năng lượng protein cục bộ (trên residue mới được thêm vào) để xử lý protein trở thành dạng gần giống với trong tự nhiên. Phân tử protein không chứa ligand và đồng kết tinh của nó sau đó được tách riêng thành 2 tệp riêng biệt được lưu dưới định dạng *.pdb và được lưu bằng MOE 2015.

Sử dụng MGLTools trên giao diện của Jupyter Notebook để mở tệp chứa cấu trúc của HIV Integrase thu được từ bước trên, sau đó thêm vào toàn bộ nguyên tử hydro có thể có của cấu trúc, bao gồm cả hydro phân cực (polar hydrogen) và hydro không phân cực (non-polar hydrogen) và điện tích Gasteiger. Tiến hành lưu lại cấu trúc cuối cùng của protein dưới dạng *.pdbqt.

2.2 Chuẩn bị ligand

Sử dụng MGLTools trên giao diện của Jupyter Notebook để mở tệp ligand Bictegravir được lưu dưới định dạng *.pdb ở bước trên và lưu tệp dưới định dạng *.pdbqt.

2.3 Xác định thông số của khoang gắn kết

Các phần mềm Docking yêu cầu phải xác định được vùng không gian gọi là Gridbox mà tại đó, các cấu dạng cần tìm phù hợp với vị trí gắn kết bị giới hạn trong vùng không gian đó trước khi tiến hành quá trình gắn kết.

Để xây dựng Gridbox, cần phải biết được kích thước và tọa độ của vùng không gian này thông qua việc quy định ba nhóm thông số chính bao gồm :

1. Kích thước Gridbox (hoặc số lượng điểm trên mỗi trục không gian X, Y, Z).
2. Khoảng cách giữa các điểm trên mỗi trục.
3. Tọa độ (x, y, z) của tâm Gridbox.

Cấu trúc Bictegravir và đồng kết tinh đã lưu dưới dạng *.pdbqt được mở trong phần mềm AutoDock Tools 1.5.7. Gridbox được xác định thông qua lệnh **Grid > Grid box**, dựa trên vị trí của đồng kết tinh 6PUW, các thông số được thiết lập sao cho hộp Gridbox bao phủ hoàn toàn 6PUW, chứa ion Mg^{2+} . Bên cạnh đó kích thước của Gridbox phải đủ rộng để ligand tham gia trong quá trình nghiên cứu có thể tự do thay đổi cấu dạng gắn kết trong khoang gắn kết của 6PUW. Thông số cụ thể được trình bày trong Bảng 2.6.

Bảng 2.2. Các thông số của khoang gắn kết Gridbox trên 6PUW

| Số điểm grid | Kích thước Gridbox | Khoảng cách giữa các điểm trên mỗi trục | Tọa độ tâm Gridbox |
|--------------|--------------------|---|--------------------|
| X = 60 | X = 22,5 | | x = 143,399 |
| Y = 60 | Y = 22,5 | 0,375 | y = 159,402 |
| Z = 60 | Z = 22,5 | | z = 177,382 |

Trong trường hợp của Autodock GPU, tệp *.gpf chứa các thông số Gridbox đã lựa chọn được tạo và lưu bằng phần mềm Autodock Tools-rc1 1.5.7. Các thông số Gridbox từ file

*gpf được đưa vào Autogrid 4.2 để thiết lập các bản đồ (maps) mô tả quá trình tương tác của các nguyên tử từng nguyên tố có trong khoang gắn kết của protein và của ligand.

2.4 Redocking

2.4.1 Phần mềm Smina, Qvina2 và Autodock Vina 1.2.3

Các phần mềm này nhận tệp đầu vào là *pdbqt (phối tử và protein sau khi đã chuẩn bị bằng MGLTools) để tiến hành gắn kết phân tử giữa protein và ligand đồng kết tinh. Tập tin kết quả thu được gồm kết quả gắn kết của tối đa 1000 cấu dạng tốt nhất cho từng ligand (*pdbqt và *pdb). Các thông số kiểm soát quá trình redocking và docking của các phần mềm được mô tả trong Bảng 2.7. Kết quả redocking được đánh giá dựa trên 3 yếu tố:

- Ái lực gắn kết thông qua năng lượng liên kết.
- Cấu dạng và vị trí gắn kết của ligand, sự chênh lệch giữa cấu dạng của ligand trước và sau khi redocking vào khoang gắn kết qua giá trị RMSD (≤ 2.0 Å)
- Các tương tác quan trọng giữa ligand đồng kết tinh sau khi redocking so với cấu trúc đồng kết tinh trong protein ban đầu.

Bảng 2.3. Thông số của các phần mềm Smina, Qvina2 và Autodock Vina 1.2.3

| Thông số | Ý nghĩa | Giá trị |
|----------------|---|---------|
| seed | Ràng buộc khởi tạo ngẫu nhiên | 42 |
| Exhaustiveness | Số lần tìm kiếm toàn cục | 100 |
| num_modes | Số lượng cấu dạng gắn kết tối đa khởi tạo | 1000 |
| cpu | Số lượng cpu sử dụng | -1 |

Quá trình docking và redocking được thực hiện trên máy tính CPU AMD Ryzen 9 3900X, hệ điều hành Ubuntu 20.04.

2.4.2 Phần mềm Vina-GPU

Các bước chuẩn bị redocking và đánh giá tương tự với Autodock Vina 1.2.3, chỉ khác các thông số được mô tả trong Bảng 2.8

Bảng 2.4. Thông số của các phần mềm Vina-GPU

| Thông số | Ý nghĩa | Giá trị |
|--------------|---|---------|
| seed | Ràng buộc khởi tạo ngẫu nhiên | 42 |
| thread | Quy mô của quá trình dock song song (docking lanes) | 8000 |
| num_modes | Số lượng cấu dạng gắn kết tối đa khởi tạo | 1000 |
| search_depth | Số lần lặp lại tìm kiếm ở mỗi docking lane | 50 |

Quá trình docking và redocking được thực hiện trên máy tính ảo Intel(R) Xeon(R) CPU @ 2.00GHz, hệ điều hành Ubuntu 18.04, GPU Tesla T4 16GB.

2.4.3 Autodock-GPU

Autodock-GPU nhận 2 tệp đầu vào là *.pdbqt (ligand) và *.map.fld để tiến hành quá trình gắn kết phân tử giữa protein và ligand đồng kết tinh. Tệp tin kết quả được trích xuất dưới dạng tệp *dlg chứa kết quả gắn kết (bao gồm cấu dạng ligand và năng lượng gắn kết, hệ số ức chế dự đoán K_i tương ứng với từng cấu dạng). Tệp *dlg được chuyển thành *.pdbqt bằng MGLTools, được dùng để đánh giá kết quả và phân tích tương tác giữa ligand đồng kết tinh với protein sau khi docking. Các thông số kiểm soát quá trình redocking và docking của Autodock GPU được mô tả trong Bảng 2.9. Kết quả redocking được đánh giá dựa trên 3 yếu tố tương tự như Autodock Vina 1.2.3.

Bảng 2.5. Thông số của các phần mềm Autodock-GPU

| Thông số | Ý nghĩa | Giá trị |
|----------|--|-----------------|
| seed | Ràng buộc khởi tạo ngẫu nhiên | 42, 42 |
| nrun | Số lần chạy thuật toán di truyền | 1000 |
| nev | Số lần đánh giá mỗi lượt chạy thuật toán di truyền | 2×10^9 |

Quá trình docking và redocking được thực hiện trên máy tính Intel(R) Xeon(R) CPU @ 2.00GHz, hệ điều hành Ubuntu 18.04, GPU Tesla T4 16GB.

2.5 Thử nghiệm hồi chứng (Retrospective control)

Quy trình xây dựng thử nghiệm hồi chứng bao gồm các bước sau:

- **Chuẩn bị protein:** cấu trúc protein 6PUW tương tự phần redocking (*.pdbqt)
- **Chuẩn bị tập hoạt tính:** thực hiện phân cụm dữ liệu và xây dựng tập hoạt tính tương tự như quá trình xây dựng mô hình pharmacophore. Điểm khác biệt là ngưỡng tương đồng được điều chỉnh thành 0,35 (ít khắc nghiệt hơn giá trị 0,3 trong mô hình pharmacophore) và chọn ra 30 chất từ các phân cụm tạo thành. Trường hợp tập hoạt tính chưa bao gồm ligand đồng kết tinh Bictegravir, nghiên cứu đưa thêm Bictegravir vào tập hoạt tính. Các chất được lưu dưới định dạng *.smi.
- **Chuẩn bị tập môi nhử:** các chất từ tập hoạt tính (*.smi) áp dụng mô hình học sâu Deepcoy để tạo các môi nhử với tỷ lệ active : decoy = 1:50. Tập môi nhử (*.smi) vừa được tạo thành sẽ được chuyển sang cấu trúc 2D và tối thiểu hoá năng lượng bằng thư viện RDKit. Sau đó, sử dụng MGLTools trên giao diện của Jupyter Notebook để tiến hành chuẩn bị ligand.
- **Gắn kết phân tử:** tiến hành tương tự redocking. Tuy nhiên, để tối ưu hoá tốc độ, nghiên cứu sử dụng thêm batch-docking (Autodock vina 1.2.3 và Autodock-GPU) để tiến hành gắn kết song song nhiều phối tử cùng một thời điểm. Tập tin thu được (*.pdbqt và *.pdb) chứa thông tin của 1000 cấu dạng, nhưng nghiên cứu chỉ dùng cấu dạng tốt nhất để tiến hành thử nghiệm hồi chứng. Thông tin kết quả gắn kết của từng ligand, bao gồm điểm số Docking và các tương tác với acid amin trong khoang được trích xuất tự động vào tệp *.csv.
- **Phân tích kết quả thử nghiệm hồi chứng:**
 - Chuẩn hoá điểm số docking bằng công cụ Min Max Scaler.
 - Xây dựng đường cong ROC với từng ngưỡng Docking.
 - Tính giá trị diện tích dưới đường cong (AUC).
 - So sánh và chọn ra phần mềm phù hợp cho việc docking.

- Xác định điểm gãy của đường cong bằng đại lượng G-mean.

3 TÀI LIỆU THAM KHẢO

111. Oshiro CM, Kuntz ID, Dixon JS. Flexible ligand docking using a genetic algorithm. *Journal of computer-aided molecular design*. 1995;9(2):113-30. doi:10.1007/bf00124402
112. Fan J, Fu A, Zhang L. Progress in molecular docking. *Quantitative Biology*. 2019;7(2):83-89.
113. Kitchen DB, Decornez H, Furr JR, Bajorath J. Docking and scoring in virtual screening for drug discovery: methods and applications. *Nature reviews Drug discovery*. 2004;3(11):935-49. doi:10.1038/nrd1549
114. Yadava U. Search algorithms and scoring methods in protein-ligand docking. *Endocrinol Int J*. 2018;6(6):359-367.
115. Bender BJ, Gahbauer S, Luttens A, et al. A practical guide to large-scale docking. *Nature protocols*. 2021;16(10):4799-4832. doi:10.1038/s41596-021-00597-z
116. Akosa, Josephine. Predictive accuracy: A misleading performance measure for highly imbalanced data. *Proceedings of the SAS global forum*. Vol. 12. 2017.
117. Chen Y-CJ. Beware of docking! *Trends in pharmacological sciences*. 2015;36(2):78-95.
118. Koebel MR, Schmadeke G, Posner RG, Sirimulla S. AutoDock VinaXB: implementation of XBSF, new empirical halogen bond scoring function, into AutoDock Vina. *Journal of cheminformatics*. 2016;8(1):1-8.
119. Koes DR, Baumgartner MP, Camacho CJ. Lessons learned in empirical scoring with smina from the CSAR 2011 benchmarking exercise. *Journal of chemical information and modeling*. 2013;53(8):1893-904. doi:10.1021/ci300604z
120. Alhossary A, Handoko SD, Mu Y, Kwoh C-K. Fast, accurate, and reliable molecular docking with QuickVina 2. *Bioinformatics*. 2015;31(13):2214-2216.

121. Eberhardt J, Santos-Martins D, Tillack AF, Forli S. AutoDock Vina 1.2. 0: New docking methods, expanded force field, and python bindings. *Journal of Chemical Information Modeling*. 2021;61(8):3891-3898.
122. Shidi T, Ruiqi C, Mengru L, et al. Accelerating AutoDock VINA with GPUs. 2022;