Supplementary Information (SI) for Physical Chemistry Chemical Physics.

Title: Alchemical Space Exploration in Drug Design via Stochastic Expanding Boundary Optimization Search

AUTHOR NAMES:

Konstantina Vaitsi,[a] *Vasilios S. Melissas*[b] *and Georgios C. Boulougouris**[a,c]

AUTHOR ADDRESS

[a]Laboratory of Computational Physical Chemistry, Department of Molecular Biology and Genetics, Democritus University of Thrace, GR-681 00 Alexandroupoli, Greece

[b]Department of Chemistry, University of Ioannina, GR-451 10 Ioannina, Greece

[c]Institute of Nanoscience and Nanotechnology, National Center for Scientific Research "Demokritos", GR-153 41 Agia Paraskevi Atikis, Greece

*To whom correspondences should be addressed

GCB: gbouloug@mbg.duth.gr

Drug design, Docking, binding affinity, Materials' design, alchemical transformations.

Supplementary Info

S.1.: SEBOS installation Instructions:

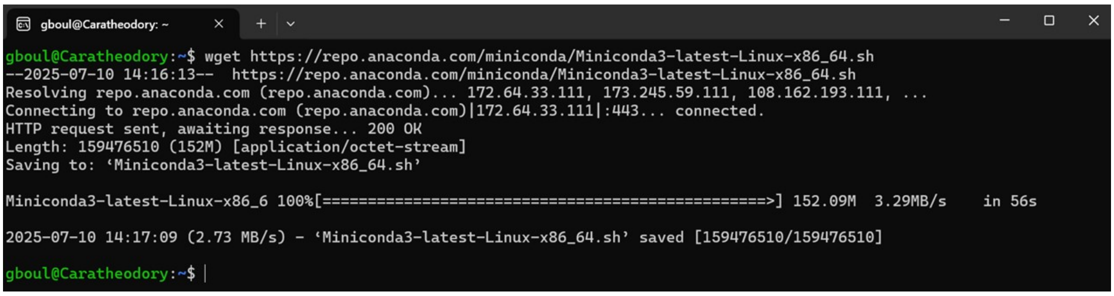
Steps of acquiring the prerequisites for a clean installation using the install_molpher.sh script :

conda , gcc , g++ ,cmake , molpher, linux/wsl.

Creates a conda environment with all module prerequisites.

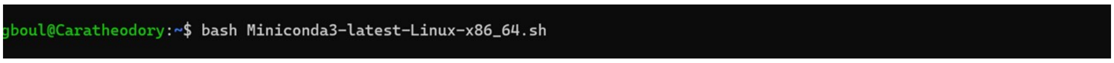
- **Step 1 :** Install miniconda :

$wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh



```
gboul@Caratheodory:~$ wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
--2025-07-10 14:16:13--  https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
Resolving repo.anaconda.com (repo.anaconda.com)... 172.64.33.111, 173.245.59.111, 108.162.193.111, ...
Connecting to repo.anaconda.com (repo.anaconda.com)|172.64.33.111|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 159476510 (152M) [application/octet-stream]
Saving to: ‘Miniconda3-latest-Linux-x86_64.sh’

Miniconda3-latest-Linux-x86_6 100%[=============================================>] 152.09M  3.29MB/s    in 56s

2025-07-10 14:17:09 (2.73 MB/s) - ‘Miniconda3-latest-Linux-x86_64.sh’ saved [159476510/159476510]

gboul@Caratheodory:~$ |
```

$bash Miniconda3-latest-Linux-x86_64.sh

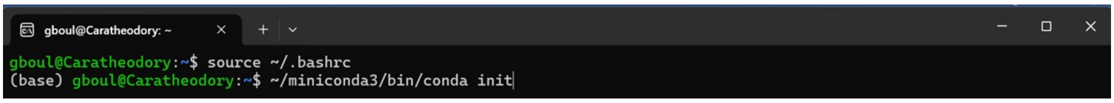


```
gboul@Caratheodory:~$ bash Miniconda3-latest-Linux-x86_64.sh
```

Activate conda :

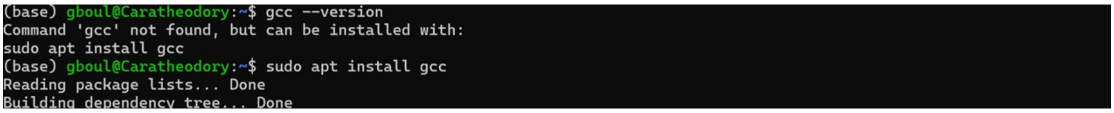
$~/miniconda3/bin/conda init

$source ~/.bashrc

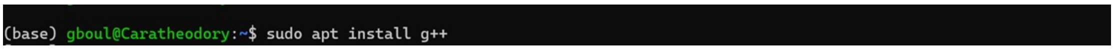


```
gboul@Caratheodory:~$ source ~/.bashrc
(base) gboul@Caratheodory:~$ ~/miniconda3/bin/conda init
```

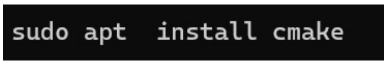
- **Step 2 :** Check/install for C and C++ compilers and cmake .



```
(base) gboul@Caratheodory:~$ gcc --version
Command 'gcc' not found, but can be installed with:
sudo apt install gcc
(base) gboul@Caratheodory:~$ sudo apt install gcc
Reading package lists... Done
Building dependency tree... Done
```

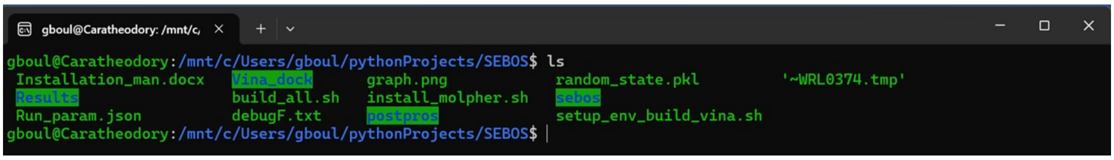


```
(base) gboul@Caratheodory:~$ sudo apt install g++
```



```
sudo apt  install cmake
```

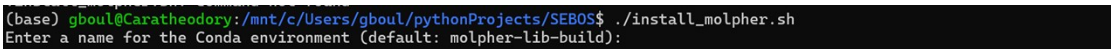
- **Step 3.:** Extract and Navigate to the SEBOS directory.



```
gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$ ls
 Installation_man.docx   Vina_dock     graph.png         random_state.pkl         '~WRL0374.tmp'
 Results                 build_all.sh  install_molpher.sh  sebos
 Run_param.json          debugF.txt    postpros          setup_env_build_vina.sh
gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$ |
```

- **Step 4. :** Download and compile molpher python module library. create a conda envaroment that access the library :

  $./install_molpher.sh



```
(base) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$ ./install_molpher.sh
Enter a name for the Conda environment (default: molpher-lib-build):
```

Add the following envaromental variable to your ~/.bashrc or ~/.zshrc as instructed by the script.

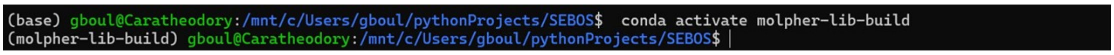
Example (replace xxx and yyy based on your installation)

export LD_LIBRARY_PATH=/xxx/molpher-lib/dist/lib/:/yyy/miniconda3/envs/molpher-lib-build/lib/

export PYTHONPATH=/xxx/molpher-lib/dist/lib/python3.9/site-packages/mol:/xxx/molpher-lib/dist/lib/python3.9/site-packagesmolpher-0.0.0b4.dev1-py3.9-linux-x86_64.egg

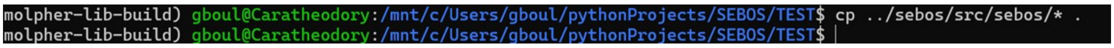
Activate the molpher environment

$conda activate molpher-lib-build



```
(base) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$  conda activate molpher-lib-build
(molpher-lib-build) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$ |
```

Copy Source files to a running directory :



```
molpher-lib-build) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS/TEST$ cp ../sebos/src/sebos/* .
molpher-lib-build) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS/TEST$ |
```

- Run SEBOS python code:

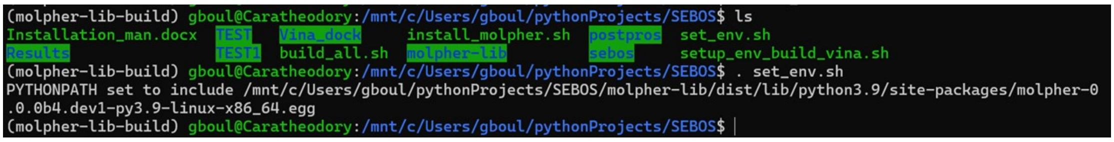
$python SEBOS.py

Note please be sure that the LD_LIBRARY_PATH and PYTHONPATH envaromental variables is set correctly. Those are set during initial setup but have to be manualy added in .bashrc of set at every terminal. You can use the set_env.sh script to add the appropriate PYTHONPATH environment variable . You can use either the comments

$. set_env.sh

or

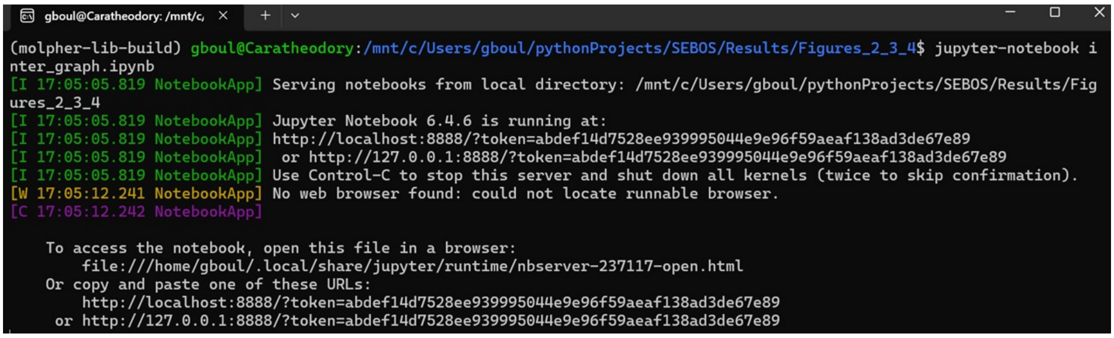
$ source set_env.sh



```
(molpher-lib-build) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$ ls
Installation_man.docx  TEST   Vina_dock     install_molpher.sh  postpros  set_env.sh
Results                TEST1  build_all.sh  molpher-lib         sebos     setup_env_build_vina.sh
(molpher-lib-build) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$ . set_env.sh
PYTHONPATH set to include /mnt/c/Users/gboul/pythonProjects/SEBOS/molpher-lib/dist/lib/python3.9/site-packages/molpher-0
.0.0b4.dev1-py3.9-linux-x86_64.egg
(molpher-lib-build) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$ |
```

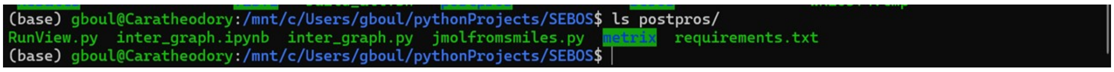
- Use an interactive environment to View and analyses results at a Post process stage:

  $jupyter-notebook inter_graph.ipynb



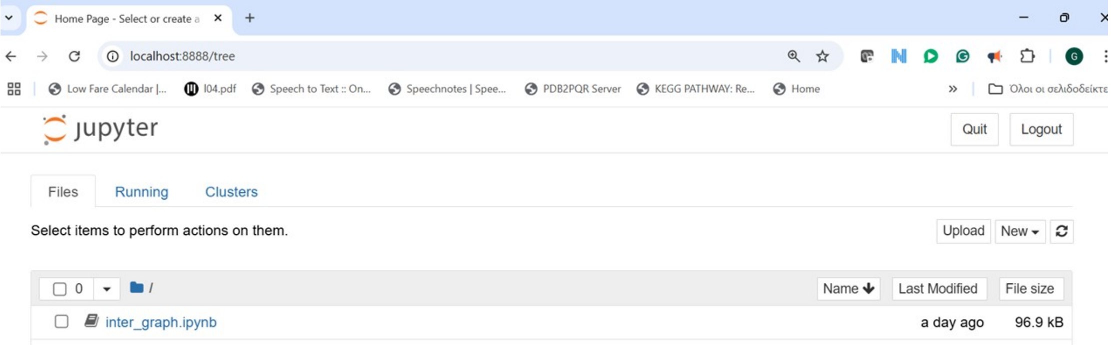
```
(molpher-lib-build) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS/Results/Figures_2_3_4$ jupyter-notebook i
nter_graph.ipynb
[I 17:05:05.819 NotebookApp] Serving notebooks from local directory: /mnt/c/Users/gboul/pythonProjects/SEBOS/Results/Fig
ures_2_3_4
[I 17:05:05.819 NotebookApp] Jupyter Notebook 6.4.6 is running at:
[I 17:05:05.819 NotebookApp] http://localhost:8888/?token=abdef14d7528ee939995044e9e96f59aeaf138ad3de67e89
[I 17:05:05.819 NotebookApp]  or http://127.0.0.1:8888/?token=abdef14d7528ee939995044e9e96f59aeaf138ad3de67e89
[I 17:05:05.819 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[W 17:05:12.241 NotebookApp] No web browser found: could not locate runnable browser.
[C 17:05:12.242 NotebookApp]

    To access the notebook, open this file in a browser:
        file:///home/gboul/.local/share/jupyter/runtime/nbserver-237117-open.html
    Or copy and paste one of these URLs:
        http://localhost:8888/?token=abdef14d7528ee939995044e9e96f59aeaf138ad3de67e89
     or http://127.0.0.1:8888/?token=abdef14d7528ee939995044e9e96f59aeaf138ad3de67e89
```

Note : The necessary postprocess files can be found in the directory postpros/ :



```
(base) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$ ls postpros/
RunView.py  inter_graph.ipynb  inter_graph.py  jmolfromsmiles.py  metrix  requirements.txt
(base) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$
```

In no browser has been installed use localhost:8888

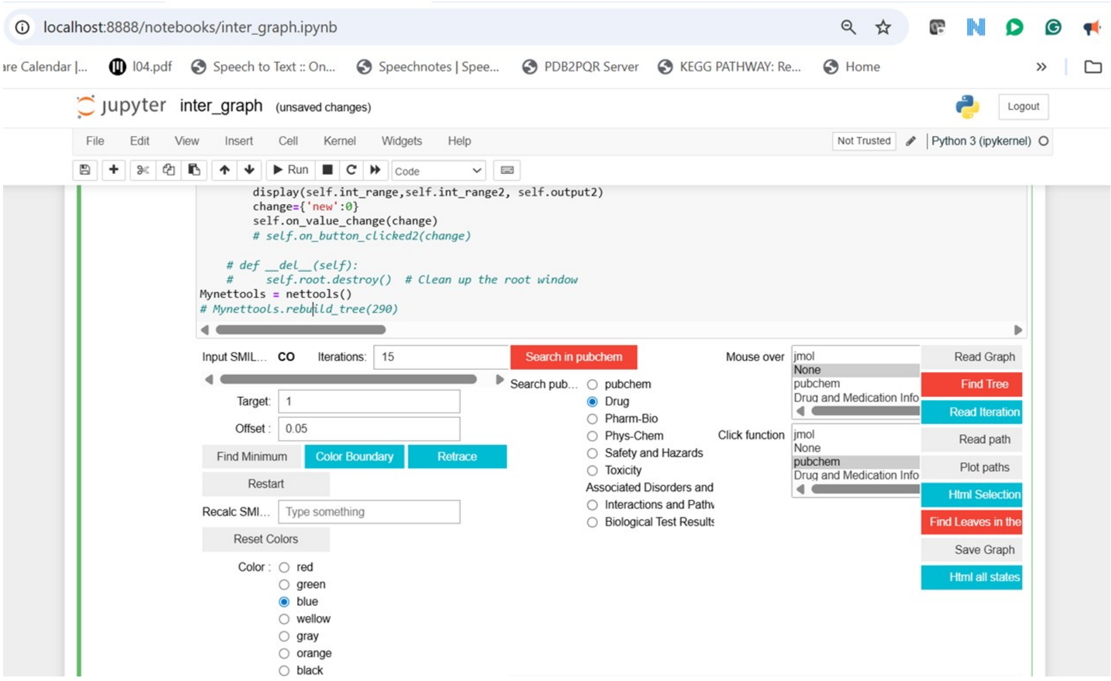


Select items to perform actions on them.

□ 0 /

□ inter_graph.ipynb a day ago 96.9 kB

Select to open the inter_graph.ipynb file , and run using shift+enter on the cell with the code:



```
        display(self.int_range,self.int_range2, self.output2)
        change={'new':0}
        self.on_value_change(change)
        # self.on_button_clicked2(change)

    # def __del__(self):
    #     self.root.destroy()  # Clean up the root window
Mynettools = nettools()
# Mynettools.rebuild_tree(290)
```

Input SMIL... CO Iterations: 15

Target: 1

Offset : 0.05

Recalc SMI... Type something

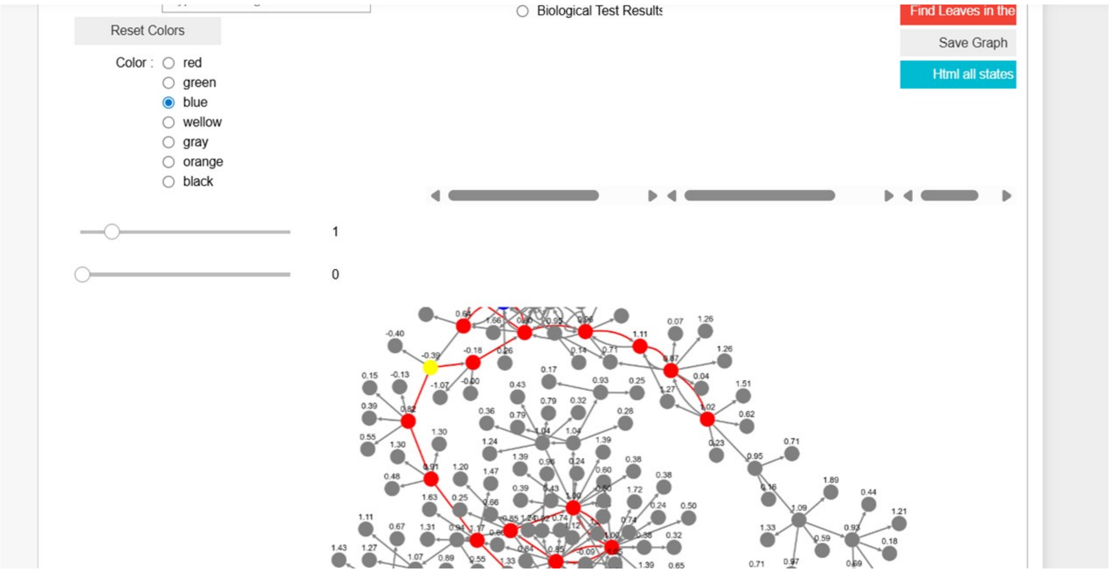
Color : ○ red ○ green ● blue ○ wellow ○ gray ○ orange ○ black

Search pub... ○ pubchem ● Drug ○ Pharm-Bio ○ Phys-Chem ○ Safety and Hazards ○ Toxicity Associated Disorders and ○ Interactions and Pathw ○ Biological Test Results

Mouse over: jmol, [✓ None], pubchem, Drug and Medication Info

Click function: jmol, None, [✓ pubchem], Drug and Medication Info

Use Cytoscape and the additional futures provided in the ipynb file to analyze the resulting network.



Color : ○ red ○ green ● blue ○ wellow ○ gray ○ orange ○ black

1

0

- How to remove the environment (xxx the path used to install the molpher library during step 4):

$conda env remove --name molpher-lib-build

$rm -rf /xxx/molpher-lib/

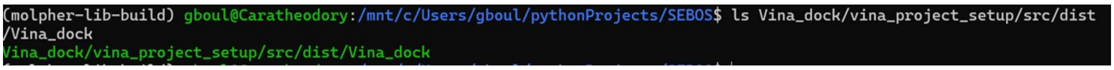
jupyter-notebook inter_graph.ipynb

SEBOS installation Instructions including Autodock vina support:

Run build_all.sh, instead of the install_molpher.sh that runs as part of the build_all.sh **(and press Enter / select yes in the prompts.)**

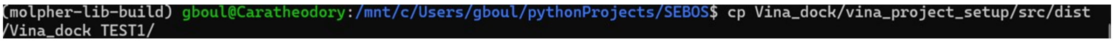
$ ./build_all.sh

This will automatically create 3 conda environments including the molpher-lib-build , download all necessary files and create an executable Vina_doc (using pyinstaller) that has to be pleased in the working directory where the python code runs.



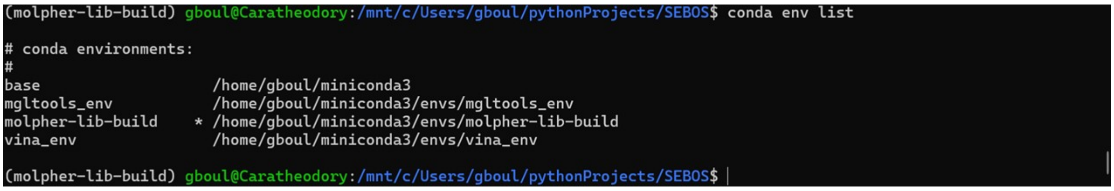
```
(molpher-lib-build) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$ ls Vina_dock/vina_project_setup/src/dist
/Vina_dock
Vina_dock/vina_project_setup/src/dist/Vina_dock
```

Copy the Vina_dock executable to your working directory :



```
(molpher-lib-build) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$ cp Vina_dock/vina_project_setup/src/dist
/Vina_dock TEST1/
```

List of the conda environments created during the installation process :



```
(molpher-lib-build) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$ conda env list

# conda environments:
#
base                     /home/gboul/miniconda3
mgltools_env             /home/gboul/miniconda3/envs/mgltools_env
molpher-lib-build     *  /home/gboul/miniconda3/envs/molpher-lib-build
vina_env                 /home/gboul/miniconda3/envs/vina_env

(molpher-lib-build) gboul@Caratheodory:/mnt/c/Users/gboul/pythonProjects/SEBOS$
```

Remove environments :

```
$conda env remove --name mgltools_env

$conda env remove --name vina_env

$conda env remove --name molpher-lib-build
```

manually delete the molpher-lib directory : $rm -rf /xxx/molpher-lib/

remove/reset environmental variables in the .bashrc (if manual changes have been made)

Note : An optional step in the workflow involves using PyInstaller to create standalone executables. PyInstaller itself is licensed under the GNU General Public License v2 (GPLv2).

◈ No PyInstaller-generated binaries are distributed as part of this project.

◈ Users are responsible for generating executables locally using their own PyInstaller installation.

If you choose to create and redistribute executables produced with PyInstaller, you must ensure that you comply with the terms of the GPLv2 license for any such distributed binaries.

S.2.: Replace the “greedy” approach of exploring network with the solution of a Master Equation:

The proposed **S**tochastic **E**xpanding **B**oundary **O**ptimization **S**earch, S.E.B.O.S., algorithm can be envisioned as the limit of a “greedy” exploration of an ever-expanding stochastic network for which a physics inspired approach based on the recently proposed Event Horizon Kinetic Monte Carlo, EH-KMC, approach could have been used instead.

Such an implementation would require the definition of an “effective rate” for each alchemical change and an “effective temperature”. A straight forward approach of such a physically inspired network in our examples of alchemical changes could have been implemented through amphidromous effective reactions definition between each pair of nodes connected by an alchemical transformation. We can use the objective function F(S_i), defined in equations (1) and (2) of the main article, in a direct analogy of a potential energy function in physical systems and create a sampling scheme, which at equilibrium will visit each node with probability proportional to $p_i^{eq} \propto e^{-\beta F(S_i)}$, where β plays the role of the inverse effective temperature in appropriate units for a dimensionless $\beta F(S_i)$. Such an “effective temperature” could be used to lead the system closer to a local minimum or to allow for “exploration” of structures over local minima via an annealing procedure.

A sufficient but not necessary approach of achieving sampling of each node with $p_i^{eq} \propto e^{-\beta F(S_i)}$ could be based on flux balance reactions, which is equivalent to the detailed balance condition in Monte Carlo. Under the flux balance condition for each edge connecting nodes *i* and *j,* two rate constants for the effective amphidromous reactions should obey equation s1.

$$e^{-\beta F\left(S_i\right)} k_{i\to j} = e^{-\beta F\left(S_j\right)} k_{j\to i} \qquad \text{s1}$$

Equation s1 provides a relative freedom in defining the rate constants $k_{i\rightarrow j}$ and $k_{j\rightarrow i}$. If a "chemical reaction analog" is to be introduced, a preexponential factor *c* should provide a time scale for the amphidromous events, whereas the additional notion of a barrier height, F^*_{ij}, has to be greater or equal to the maximum of the objective functions $F(S_i)\ and\ F(S_j), i.e.: F^*_{ij} \geq \max\left(F(S_i), F(S_j)\right)$.

$$k_{i\rightarrow j} = ce^{-\beta\left(F^*_{ij} - F(S_i)\right)},\ k_{j\rightarrow i} = ce^{-\beta\left(F^*_{ij} - F(S_j)\right)} \qquad \text{s2}$$

The preexponential factor *c* and the barrier heights F^*_{ij} are allowed to adopt other values for each set of amphidromous reactions, thus allowing the introduction of a "bias" in the search algorithm without altering the relative observation probability of each node. Although defining the effective rates using equation s2 will result in a valid scheme, however it does not provide the ability to distinguish between alternative *j* states based on the objective function, therefore, in order to sustain the ability to have higher fluxes towards states with lower objective function values, it is preferably used the rate definition of equation s3, which also satisfies the flux balance in equation s1. Namely :

$$k_{i\rightarrow j} = ce^{-\beta\left(F(S_j)\right)}, k_{j\rightarrow i} = ce^{-\beta\left(F(S_i)\right)} \qquad \text{s3}$$

It should be noted that both equations s2 and s3 will result in Kinetic Monte Carlo simulations, where the observation probability of each state is proportional to $e^{-\beta F(S_i)}$.

Given the effective rates $k_{i\rightarrow j}$, $k_{j\rightarrow i}$ for each alchemical event, we may apply the recently proposed Event Horizon Kinetic Monte Carlo(Boulougouris 2024) approach, defining a set of "boundary" states *G* with absorbing boundary conditions. The set of boundary states *G* consists of nodes as a result of alchemical changes of an ever-expanding set of "explored" states *E*. As in the S.E.B.O.S. approach, the set of "explored" states could be initialized to include only the initial node, and via an iterative process expands and incorporates at each step a boundary node. Following the methodology introduced in Event Horizon Kinetic Monte Carlo, the selection of the boundary node can be based on the analytical evaluation of the flux, $f_j(t)$, towards each boundary state *j*. Given the transition rate $k_{i\to j}$ for all *i's* in *E* and *j's* in $E \cup G$, we may write the master equation for the observation probabilities, $Q_i(t)$, of the system in each of the states *i* in the set *E*, at time t, in the form of Eq. s4.

$$\frac{\partial Q_i(t)}{\partial t} = \sum_{\substack{j\in E \\ i\neq j}} Q_j(t) k_{j\to i} - Q_i(t) \sum_{\substack{j\in E\cup G \\ i\neq j}} k_{i\to j} \ \forall i \in E \qquad \text{s4}$$

The flux towards the boundary can then be written as a sum of contributions for each one of the boundary states in equation s5,

$$f_j(t) = \sum_{i\in E} Q_i(t) k_{i\to j}\,, \forall\, j \in G \qquad \text{s5}$$

allowing for the stochastic expansion of the explored state *E* through selection of state *j* from the set of boundary state proportional to $\propto f_j(t) \Big/ \sum_{j\in G} f_j(t)$.

Since we are not interested in following the exact dynamics, we may approximate the probability $Q_i(t)$ as

$$Q_i(t) \cong \frac{e^{-\beta\left(F(S_i)\right)}}{\sum_{k\in E} e^{-\beta\left(F(S_k)\right)}}$$

the equilibrium probability for the set of "explored" states. By combining the minimal "barrier" and the "local equilibrium" approximation, one may introduce an approximate time independent estimation of the flux towards each "boundary" state *j*.

$$f_j \cong \sum_{i \in E} \frac{e^{-\beta\left(F\left(S_i\right)\right)}\, e^{-\beta\left(F\left(S_j\right)\right)}}{\sum_{k \in E} e^{-\beta\left(F\left(S_k\right)\right)}}, \forall j \in G \qquad \text{s6}$$

Thus, there is no need to solve for the master equation, s4, which may easily become computationally intensive as the size of the network increases.

Equation s6 provides an alternative selection scheme for the expansion of the local network, where we select stochastically one of the "boundary" states based on the probability value, $p_{\to S_j}$, according to equation s7.

$$p_{\to S_j} = \frac{f_j}{\sum_{l \in G} f_l} \cong \frac{\sum_{i \in E} \frac{e^{-\beta\left(F\left(S_i\right)\right)}\, e^{-\beta\left(F\left(S_j\right)\right)}}{\sum_{k \in E} e^{-\beta\left(F\left(S_k\right)\right)}}}{\sum_{l \in G} \sum_{i \in E} \frac{e^{-\beta\left(F\left(S_i\right)\right)}\, e^{-\beta\left(F\left(S_l\right)\right)}}{\sum_{k \in E} e^{-\beta\left(F\left(S_k\right)\right)}}}, \forall j \in G \qquad \text{s7}$$

It can be easily verified that as the effective temperature drops (and *β* increases), $p_{\to S_j}$ tends to denote the state with minimal $F(S_j)$, resulting in the "greedy" approach proposed in S.E.B.O.S. On the other hand, as *β* decreases, expansion of the local network allows nodes' exploration with higher objective function values.

To illustrate the influence of the "effective temperature" in the selection process of "boundary" states through equation s7, we consider the example presented in Figure 1 of the main article.

In detail, Figures S1 and S2 present the "temperature dependence" of the selection probability $p_{\to S_j}$ for the two initial steps of the expansion process. In the first step, the set of "explored" states consists of a single state {S_1} with $g(S_1)$=-0.392, and a boundary set of states

{S_2, S_3, S_4, S_5} with $g(S_2)$=-0.001, $g(S_3)$=-0.666, $g(S_4)$=-0.825 and $g(S_5)$=0.263. Since the search target is set to 1, the objective function values (provided by Eq. 2 of the main article), are calculated as F(S_1)= 1.94, F(S_2)= 1.00, F(S_3)=2.78, F(S_4)=3.33 and F(S_5)=0.543 . As it can readily be seen in Figure S1, low values of *β* will result in selection of each state with equal probability regardless of the objective function value, whereas larger values of *β* (lower effective temperatures) will preferably select states with lower objective function values. Similarly, for the second state of the expansion shown in figure S2, higher values of *β* results in selection of the state adopting the lowest objective function value. Similarly, for each step of the expansion process, low values for the effective temperature are expected to drive the system towards states with low objective functions values leading the system to the target.

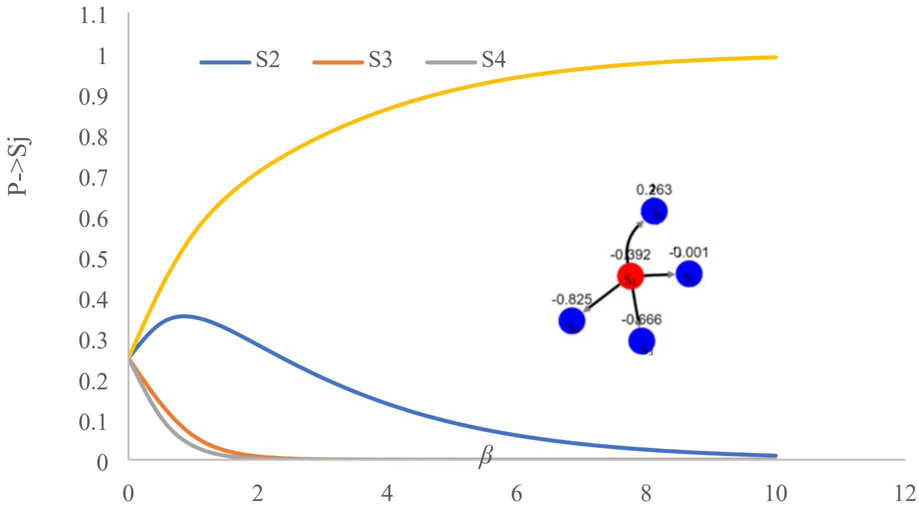


Figure S1: The “temperature effect” for the expansion of the “explored” states set at the initial step presented in Figure 1 of the main article, while equation s7 holds for the EH-MC approach and $g(S_1)$=-0.392 ,$g(S_2)$=-0.001, $g(S_3)$=-0.666, $g(S_4)$=-0.825 and $g(S_5)$=0.263 as shown in the inset.

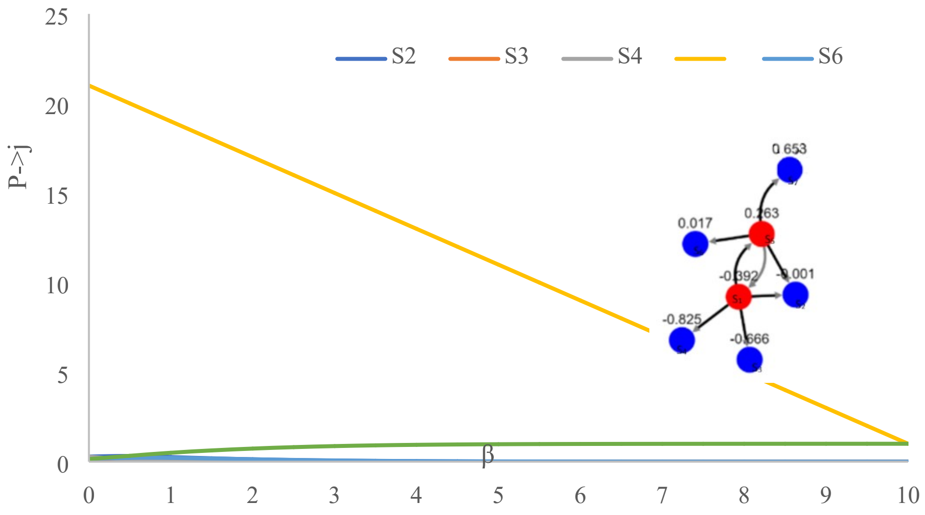


Figure S2: The “temperature effect” for the expansion of the “explored” states set of (S_1,S_5) at the second step presented in Figure 1 of the main article, while equation s7 holds for the EH-MC approach and *g(S_1)*=-0.392 ,*g(S_2)*=-0.001, *g(S_3)*=-0.666, *g(S_4)*=-0.825, *g(S_5)*=0.263, *g(S_6)*=0.017 and *g(S_7)*=0.653 as shown in the inset.

For most practical reasons, selection of the boundary state with the lowest objective function value, as in the S.E.B.O.S. approach, would be preferable, while implementing equation s7 for the stochastic network expansion is suggested only in cases where a global optimization via annealing (increasing and decreasing the effective “temperature”) is attempted.

S.3.: illustrative examples for the metrics listed in table 4:

In the H.B. analysis, presented in Table 4, the majority of ligands exhibits an increase in H-bonds contribution for the final ligand compared to the initial one. In Figure S2, we present the example of carmofur displaying a single hydrogen bond, whereas an ensuing molecule was able to form two.

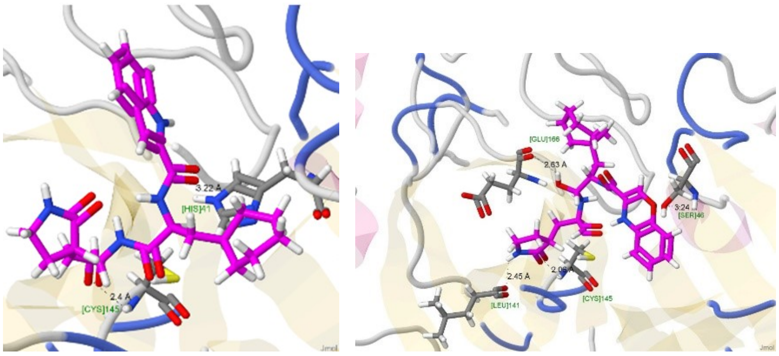


Figure S3. Comparison of hydrogen bonding interactions between the initial ligand 11a (left) and the final one (right). Atoms involved in the hydrogen bonding were calculated with the help of the smina module and denoted hydrogen bond distances, Å, and interacting residues are accordingly labeled.

According to the APBS analysis in Table 4, most of the ligands unveil an increase in the electrostatic interactions, as expected due to its key role in docking score's buildup. Figure S4 visualizes the electrostatic potential surface produced by APBS for the initial osajin molecule and its final ligand.

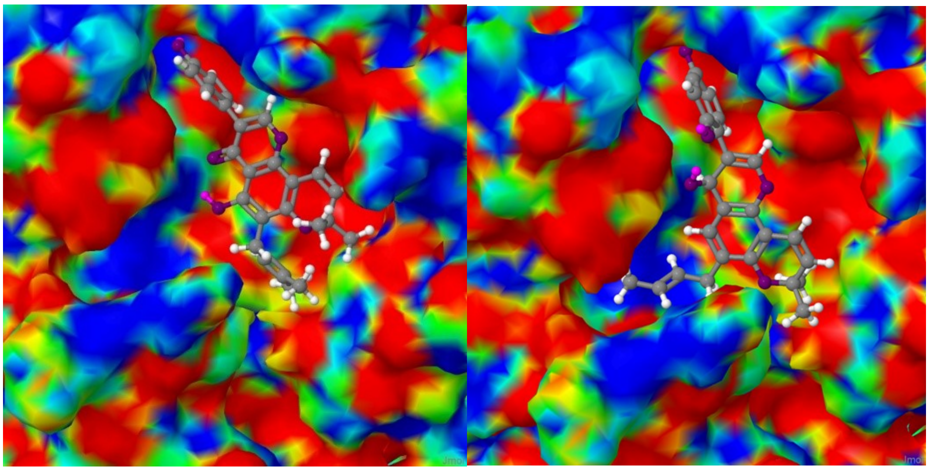


Figure S4. Comparison of electrostatic interactions between the initial ligand osajin (left) and the final ligand (right) within the same protein binding site. The electrostatic potential surface was visualized using the APBS map.dx file and Jmol. The potential

ranges from negative (red) to positive (blue), illustrating charge complementarity between each ligand and the binding pocket. Ligands are shown in stick representation, with purple atoms indicating negatively charged atoms (N, F, O) and pink atoms indicating polar hydrogens.

The N.H.R. analysis depicts the ligand's engagement with less or more hydrophobic protein residues (green ones), as shown in Figure S5.

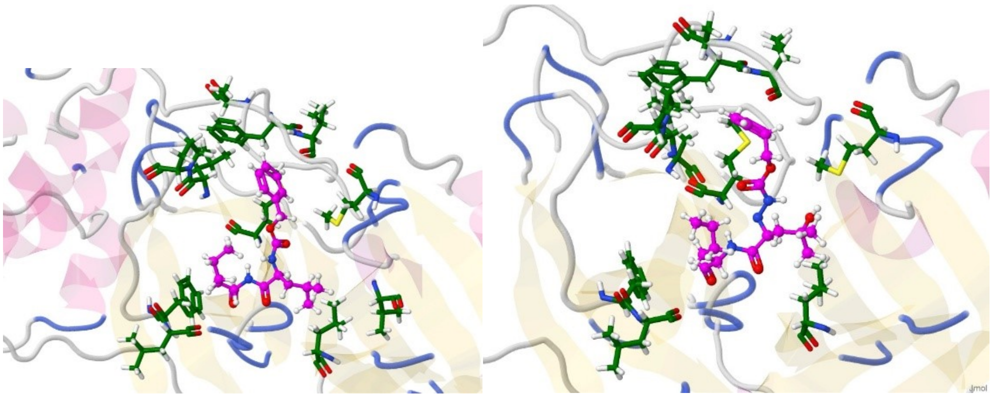


Figure S5. Comparison of hydrophobic interactions between the initial ligand calpeptin (left) and the final ligand (right) within the same protein binding site. The protein backbone is shown in cartoon representation, and hydrophobic residues are highlighted in green. Ligands are shown in magenta.

The B.S.A. metric reveals ligand's shielding, of the protein pocket. Figure S6 indicates the alteration of the B.S.A. metric between the initial digitoxin drug and the final species.

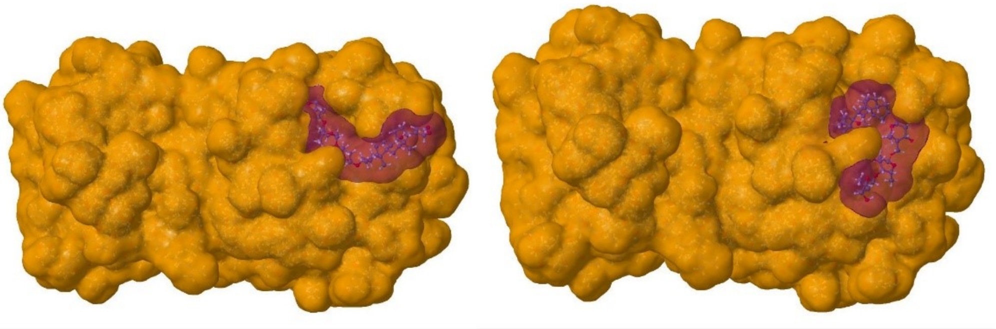


Figure S6. Comparison of buried surface area upon binding for the initial ligand digitoxin (left) and the final one (right). The protein is shown as an orange SASA surface, whereas the ligand's surface is highlighted in purple.

Finally, the Flexibility Index indicates that the majority of the finale ligands showed a decrease in this metric suggesting that the ligands became less flexible, which aligns with Vina's docking optimization to reduce entropy penalties.

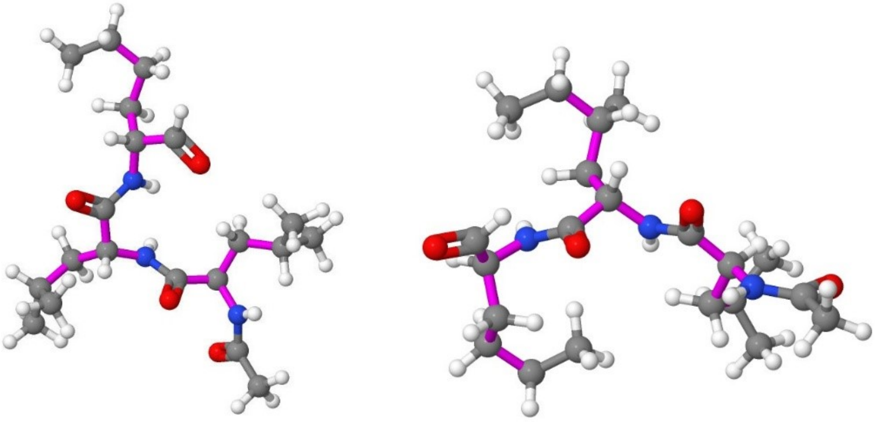


Figure S7. Comparison of ligand's flexibility between the initial drug calpain inhibitor I (left) and the final molecule (right), defined as the ratio of rotatable bonds to heavy atoms. Rotatable bonds are highlighted in magenta.

As it can be seen from Figure S7 the flexibility index maybe overestimated in cases where the AutoDock tools can not recognize the presence of conjugate structure.