

DOI:10.12113/202205009

面向虚拟筛选的 GPU 加速的分子对接方法

胡海峰^{1*}, 王领悦¹, 唐诗迪², 胡鸣珂³, 吴建盛²

(1.南京邮电大学 通信与信息工程学院,南京 210023;2.南京邮电大学 地理与生物信息学院,南京 210023;
3.南京林业大学 经济管理学院,南京 210037)

摘要:虚拟筛选是在计算机上对化合物分子进行模拟预筛选,找出容易和药物靶标结合的小分子(配体),从而降低实际实验测试次数,提高药物先导化合物的发现效率。常用的分子对接软件可以用于基于结构的虚拟筛选,寻找配体与靶标的最佳的作用模式和结合构象,并通过打分函数来筛选出潜在的配体。现有的对接软件如 AutoDock Vina 等在分子对接过程中需要耗费大量时间和计算资源,特别是面对大规模分子对接时,过长的筛选时间不能满足应用需求,因此,本文在最高效的 QVina2 对接软件基础上,提出一种基于 GPU 的 QVina 2 并行化方法 QVina2-GPU,利用 GPU 硬件高度并行体系加速分子对接。具体包括增加初始化分子构象数量,以扩展蒙特卡罗的迭代局部搜索中线程的并行规模,增加蒙特卡罗的迭代搜索的广度以减少每次蒙特卡罗迭代搜索深度,并利用 Wolfe-Powell 准则改进局部搜索算法,提高了对接精度,进一步减少蒙特卡罗迭代搜索深度,最后,在 NVIDIA Geforce RTX 3090 平台上在公开的配体数据库上验证了 QVina2-GPU 的性能,实验表明在保证分子对接精度的基础上,我们提出的 QVina2-GPU 对 Qvina2 的平均加速比达到 5.18 倍,最大加速比达到 12.28 倍。

关键词:虚拟筛选;分子对接;GPU;蒙特卡罗算法

中图分类号:R91 文献标志码:A 文章编号:1672-5565(2023)03-206-12

GPU accelerated molecular docking method for virtual screening

HU Haifeng^{1*}, WANG Lingyue¹, TANG Shidi², HU Mingke³, WU Jiansheng²

(1.School of Communications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210023, China;
2.School of Geographic and Biologic Information, Nanjing University of Posts and Telecommunications, Nanjing 210023, China;
3.College of Economics and Management, Nanjing Forestry University, Nanjing 210037, China)

Abstract: Virtual screening exploits computational platforms to simulate pre-screening in which molecules (ligands) that can readily bind to receptors are identified, reducing the number of drug tests and improving the efficiency of drug lead compounds discovery. Some commonly-used docking software have been applied to structure-based virtual screening to find the proper binding conformations and affinities. Specifically, the scoring function is employed to identify potential binders. However, the existing docking software such as AutoDock Vina are still time-consuming and computationally expensive, especially for larger scale dockings where longer execution time compromises the performance of the docking programs in the screening tasks. Hence, a GPU accelerated Molecular docking method for Virtual screening, QVina2-GPU, is proposed to parallelize the efficient QVina2 docking software, which can exploit the highly-parallel hardware architecture to speed up the process of molecule docking. Specifically, the multi-threaded parallel execution of Monte-Carlo iterative local search is enhanced by increasing initial molecule conformations. That is, the breadth of Monte-Carlo search is increased to reduce the depth of each Monte-Carlo search. Meanwhile, the rule of Wolfe-Powell is applied to upgrade the local search for better docking accuracy, which can further reduce the depth of Monte-Carlo search. Finally, QVina2-GPU was implemented with NVIDIA Geforce RTX 3090 on the public ligand database, and the experimental results demonstrate that the proposed QVina2-GPU can achieve an average acceleration of about 5.18 times and a maximum acceleration of up to 12.28 times compared to Qvina2 while guaranteeing desired docking accuracy.

Keywords: Virtual screening; Molecular docking; GPU; Monte-Carlo method

收稿日期:2022-05-20;修回日期:2022-09-23;网络首发时间:2022-10-21.

网络出版地址:<https://kns.cnki.net/kcms/detail/23.1513.Q.20221020.1420.006.html>

基金项目:国家自然科学基金(No.61571233).

*通信作者:胡海峰,男,教授,硕士生导师,研究方向:人工智能和网络信息处理.E-mail:huhf@njupt.edu.cn.

引用格式:胡海峰,王领悦,唐诗迪,等.面向虚拟筛选的 GPU 加速的分子对接方法[J].生物信息学,2023,21(3):206-217.

HU Haifeng, WANG Lingyue, TANG Shidi, et al. GPU accelerated molecular docking method for virtual screening[J]. Chinese Journal of Bioinformatics, 2023, 21(3):206-217.

虚拟筛选作为计算机辅助药物设计的实用技术,在现代药物研发中起着重要作用^[1]。虚拟筛选在计算平台上模拟药物筛选的过程,可快速从几十至上百万的小分子化合物(配体)中,筛选出有可能和受体结合的活性化合物,大大降低实际筛选小分子化合物数目,缩短药物研发周期,降低药物研发的成本^[2-3]。虚拟筛选的方法可分为基于结构的虚拟筛选和基于配体的虚拟筛选^[4]。基于结构的虚拟筛选利用分子对接技术,在已知蛋白质受体的三维结构的基础上,在结合位点处自动匹配化合物数据库中的小分子配体,用打分函数对可能的结合模式进行能量计算,最终得到小分子化合物结合活性的排名^[5-7]。而基于配体的虚拟筛选方法不需要知道目标蛋白质受体的结构,代表性的方法包括药效团模型、定量构效关系和结构相似性等方法^[8-9]。相对于基于配体的虚拟筛选,基于结构的虚拟筛选能避免活性化合物结构微小变化引起的活性改变^[10],对小分子配体的活性值预测更加准确。

常用的分子对接软件包括 AutoDock4^[11]、AutoDock Vina^[12]、AutoDock Vina 1.2.0^[13]和 Smina等^[14]可以进行基于结构的虚拟筛选,其中,AutoDock Vina 提高了结合模式预测的平均准确度,通过使用更简单的打分函数加快了搜索速度,是应用最广泛的分子对接软件。这些分子对接软件分别采用不同的搜索算法和打分函数进行分子结合活性预测,需要耗费大量时间和计算资源来完成分子对接计算,然而在实际虚拟筛选时,筛选的候选化合物越多,可以更大可能的搜索到合适小分子配体,提高先导化合物质量^[15]。因而面对大规模分子对接时,过长的筛选时间制约了分子对接软件的应用。例如 AutoDock Vina 在常规计算平台上筛选 10 亿个化合物的数据库时,每个配体的平均对接时间为 15 s,筛选所有化合物的时间大概需要 476 年,目前的分子对接软件无法满足现代药物研发的需求^[15]。

近年来研究人员针对分子对接软件对接时间过长的缺点进行了两方面改进。一方面,提出基于 AutoDock Vina 改进版本的 Qvina1 和 Qvina2 分子对接软件^[16-17],Qvina1 提出了启发式算法减少不必要的搜索次数,改进了局部搜索算法,与 AutoDock Vina 相比提高了运行速度,但对接精度方面有所下降。在此基础上,Qvina2 优化了启发式算法,提高了对接精度,与 AutoDock Vina 相比实现了 20.49 倍的最大加速,是已知 AutoDock Vina 系列中最为高效的对接软件。另一方面,虚拟筛选中的分子对接计算特别适合使用并行计算加速,GPU(Graphic processing unit)因其强大的并行处理硬件架构特别适合应用于分子对接软件^[18-19]。

GPU 具有数千个计算核心,可提供强大的计算性能,性价比较高且易于开发,并且有完善的开发标准,例如并行计算架构(CUDA)和开放运算语言(OpenCL)^[20]。近年来,考虑到分子对接软件的特点,基于 GPU 的异构体系也被应用于改进分子对接软件^[21],例如实现了 AutoDock4 的加速版本 AutoDock4-GPU^[22],在细粒度上并行了拉马克遗传算法,取得了很好的加速比。但 AutoDock4 的搜索效率很低^[13],制约了 AutoDock4-GPU 的加速性能。

本文提出一种基于 GPU 的 QVina 2 并行化方法 Qvina2-GPU,在目前 AutoDock Vina 系列中最为高效的 Qvina2 基础上,利用 GPU 硬件高度并行体系加速分子对接过程。具体包括增加初始化分子构象数量,大量线程分别对应不同的初始构象,使得每个线程独立承担蒙特卡罗搜索算法子任务,独立搜索最佳的分子构象,即通过增加蒙特卡罗的迭代搜索的广度以减少每次蒙特卡罗迭代搜索深度,并利用 Wolfe-Powell 准则改进局部搜索算法,相对于 QVina 2 中基于 Armijo 准则搜索算法,提高了对接精度,进一步减少蒙特卡罗迭代搜索深度,从而显著减少了蒙特卡罗构象搜索时间,减少了分子对接软件的整体运行时间。本文的贡献归纳如下:

1) 提出基于 GPU 的 QVina 2 并行化方法 Qvina2-GPU,设计 QVina2-GPU 异构并行架构,增加初始化分子构象数量,扩展蒙特卡罗的迭代局部搜索中线程的并行规模,每个线程独立承担蒙特卡罗搜索算法子任务,通过增加蒙特卡罗迭代搜索广度以减少每次蒙特卡罗迭代搜索深度。

2) 提出基于 Wolfe-Powell 准则的局部搜索算法,通过迭代步长的优化,改进了 QVina 2 中基于 Armijo 准则搜索算法,提高了分子对接精度,进一步减少蒙特卡罗迭代搜索深度。

3) 使用公开数据库中 140 个小分子复合物^[22]作为测试数据集,实验结果验证了在保证对接精度的条件下,我们提出的 QVina2-GPU 对 Qvina2 的平均加速比达到 5.18 倍,最大加速比达到 12.28 倍,具有很大的实际应用前景。

1 QVina-GPU 异构并行架构

QVina2-GPU 异构并行架构如图 1 所示,从图中可以看出,QVina2-GPU 整体架构由主机端和设备端组成,主机端由数据准备模块和优化输出模块组成,这部分软件运行在 CPU 平台上,主要完成环境配置、数据输入以及筛选最优分子构象的功能;设备端运行蒙特卡罗并行算法模块,利用 GPU 并行化处理

能力进行算法加速,主要思想是通过大量增加蒙特卡罗迭代搜索的初始态数量,以减少每次搜索深度,

这样在保证全局优化搜索结果一致的情况下,大量减少了分子对接软件运行时间。

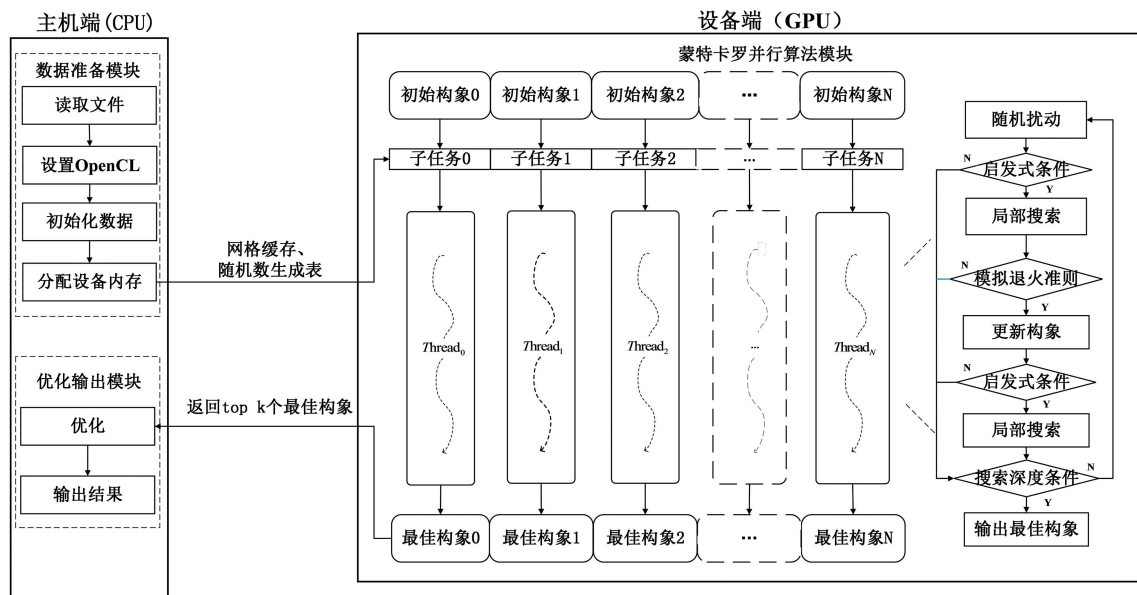


图 1 QVina2-GPU的异构并行架构

Fig.1 Heterogeneous parallel architecture of QVina2-GPU

数据准备模块包括读取文件、配置环境、初始化数据和分配设备内存四个操作,为设备端基于 GPU 的蒙特卡罗并行算法的输入做准备。

1) 读取文件模块用于读取配体和受体的格式文件和配置文件。格式文件存储受体和配体原子坐标、部分电荷和原子种类信息;配置文件主要包含对接中心、对接盒子的体积、线程数和搜索深度,其中线程数 N 是设备端子任务的数量,搜索深度 D 决定蒙特卡罗并行算法迭代次数的大小。

2) 设置 Opencl 模块用于配置 QVina2-GPU 的异构并行环境,配置环境主要包括识别并选择平台和设备,创建 OpenCL 上下文、命令队列、程序对象和内核等 OpenCL 环境。

3) 初始化数据包括生成网格缓存和随机数生成表,分别用于计算分子构象能量以及生成初始化分子构象,作为蒙特卡罗迭代搜索的初始态输入。

4) 分配设备内存模块根据设定的子任务数量 N ,将生成的网格缓存、随机数生成表、初始化分子构象分配在只读内存的设备存储器中,便于设备端读取数据。

优化输出模块对设备端基于 GPU 的蒙特卡罗并行算法的输出数据进行处理。设备端每个线程运行蒙特卡罗并行算法得到的最优分子构象分配在全局存储器中,从 N 个分子构象中选取前 k 个最佳分子构象返回给主机端的优化输出模块,该模块对 k 个最佳构象进行精细的局部搜索,并根据搜索结果

输出包含全局最优分子构象的配体文件。

2 基于 GPU 的蒙特卡罗并行算法

从图 1 可以看出设备端的基于 GPU 的蒙特卡罗并行算法主要两个部分:1) 分配 N 个蒙特卡罗迭代搜索初始态;2) GPU 子任务执行的基于 Wolfe-Powell 准则的蒙特卡罗搜索算法。

2.1 初始化蒙特卡罗迭代搜索

分子构象是指配体小分子和靶标蛋白质结合时配体的位置和姿态,蒙特卡罗搜索算法目的是在分子构象空间中找到一个最佳配体分子构象,并在此基础上通过打分函数来计算配体和靶标结合的能量值,分子构象 $C_i \in i^{N_c}$ 由一系列变量构成,具体表示为

$$C_i = [(x^i, y^i, z^i), (a^i, b^i, c^i, d^i), (\varphi_1^i, \varphi_2^i, \dots, \varphi_{N_{rot}}^i)]^T \quad (1)$$

其中, (x^i, y^i, z^i) 表示分子构象 C_i 在搜索空间中的位置, (a^i, b^i, c^i, d^i) 表示刚体方向四元数, $(\varphi_1^i, \varphi_2^i, \dots, \varphi_{N_{rot}}^i)$ 表示分子构象中可旋转键的扭转角度, N_{rot} 表示可旋转键的个数, N_c 是分子构象的维度数。

基于 GPU 的蒙特卡罗并行算法构造出 N 个初始化分子构象 $\mathcal{L} = \{C_1, \dots, C_N\}$,并利用 GPU 并行化处理能力,创建 N 个线程数,使得分子构象 $C_i \in \mathcal{L}$ 作为线程 T_i 的输入值,运行 GPU 子任务执行的蒙特卡罗算法,搜索出针对该初始分子构象 C_i 的最佳构象 C_i^* 。从上述描述可以看出,大量线程分别

对应不同的初始构象,使得每个线程独立承担蒙特卡罗搜索算法子任务,独立搜索最佳的分子构象。因而,大量的搜索线程可以实现对分子构象空间的有效搜索,可以减少单独线程蒙特卡罗搜索算法的搜索深度,在保证全局优化搜索结果一致的情况下,显著减少了分子对接软件运行时间。

2.2 子任务执行的基于 Wolfe-Powell 准则的蒙特

卡罗搜索

每个线程运行的基于 Wolfe-Powell 准则蒙特卡罗搜索算法表示为 GPU 子任务,蒙特卡罗的算法流程如图2所示,大体可以分为5步,包括随机扰动、启发式条件、局部搜索和模拟退火准则四个模块。下面以线程 T_i 处理初始分子构象 $C_i \in \mathcal{L}$ 为例说明子任务执行基于 Wolfe-Powell 准则的蒙特卡罗搜索过程。

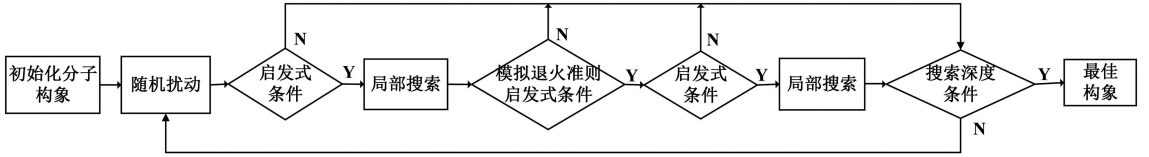


图2 基于 Wolfe-Powell 准则的蒙特卡罗搜索算法流程

Fig. 2 Flow of Monte Carlo search algorithm based on Wolfe Powell criterion

Step 1 首先对用随机抖动函数 $R(\cdot)$ 对初始分子构象 C_i 进行处理:

$$\hat{C}_i = R(C_i) \quad (2)$$

其中, $\hat{C}_i \in i^{N_c}$ 是对 C_i 中每一位特征进行随机扰动后的构象。然后,根据能量计算函数 $f(\cdot)$ 计算 \hat{C}_i 对应的和靶标结合的能量值 $E_{\hat{C}_i}$:

$$E_{\hat{C}_i} = f(\hat{C}_i) \quad (3)$$

其中,能量计算函数 $f(\cdot)$ 根据构象 \hat{C}_i 中每个原子的空间坐标计算得到分子内能量,利用三线性插值方法查找配体与受体之间的原子对的分子间能量,最后将分子间能量和分子内能量相加得到构象的能量值。

Step 2 对于随机抖动构象 \hat{C}_i ,线程 T_i 利用其他线程存储的构象 \bar{C}_i 和启发式条件来检测 \hat{C}_i ,如果不满足启发式条件就返回 step1,如果满足条件才继续执行,这样可以减少不必要的局部搜索,提高搜索效率,启发式条件定义如下:

$$\text{sign}\left(\frac{\partial f(C)}{\partial c_i} \Big|_{c=\hat{c}_i}\right) \cdot \text{sign}\left(\frac{\partial f(C)}{\partial c_i} \Big|_{c=\bar{c}_i}\right) \leq 0 \quad (4a)$$

$$\text{sign}\left(\frac{\partial f(C)}{\partial c_i} \Big|_{c=\hat{c}_i}\right) \cdot \text{sign}\left(\left[f(\hat{C}_i) - f(\bar{C}_i)\right] \left[(\hat{C}_i)_i - (\bar{C}_i)_i\right]\right) \leq 0 \quad (4b)$$

其中, $c_i \in C$ 表示构象 C_i 中的第 i 个变量, $\partial f(C)/\partial c_i|_{c=\hat{c}_i}$ 表示能量计算函数 $f(\cdot)$ 对 \hat{C}_i 中 i 个变量 $(\hat{C}_i)_i$ 的偏导, $\partial f(C)/\partial c_i|_{c=\bar{c}_i}$ 表示能量计算函数 $f(\cdot)$ 对 \bar{C}_i 的中 i 个变量 $(\bar{C}_i)_i$ 的偏导, $\text{sign}(\cdot)$ 是符号函数。公式(4a)说明能量计算函数

在抖动构象 \hat{C}_i 和构象 \bar{C}_i 处所有变量的偏导数符号相反,公式(4b)说明对于所有变量,能量计算函数在抖动构象 \hat{C}_i 的偏导数和 \hat{C}_i 相对于 \bar{C}_i 增长方向相反,上述两个条件都是说明了所有变量 $(\hat{C}_i)_i$ 和 $(\bar{C}_i)_i$ 之间一定存在静态点,而这些静态点有可能是最佳构象的位置,所以满足了上述启发式条件之一,就继续执行最佳构象的搜索过程。

Step 3 对满足启发式条件的构象 $\hat{C}_i \in i^{N_c}$,进一步利用基于 Wolfe-Powell 准则的 BFGS (Broyden Fletcher Goldfarb Shanno) 局迭代搜索算法去搜索新的分子构象,使得新构象的能量接近全局最小值。首先,初始化海森矩阵 $\mathbf{B}_0 = \mathbf{I}$, \mathbf{I} 为 $N_c \times N_c$ 的单位矩阵,初始化搜索构象 $\mathbf{x}_0 = \hat{C}_i$, BFGS 的第 $k(k = 0, 1, \dots, T)$ 次迭代过程如下:

1) 计算搜索方向 \mathbf{p}_k : $\mathbf{B}_k \mathbf{p}_k = -\nabla f(\mathbf{x}_k)$, 其中, $f(\cdot)$ 为能量计算函数, \mathbf{x}_k 和 \mathbf{B}_k 分别是第 k 步搜索构象和海森矩阵。

2) 确定基于 Wolfe-Powell 准则的步长 α_k :

$$\alpha_k = \arg \min_{\alpha \in i} f(\mathbf{x}_k + \alpha \mathbf{p}_k) \quad (5)$$

上式表达的是精确线性搜索中可接受的搜索步长 α_k ,实际执行中,我们采用非精确线性搜索中满足 Wolfe-Powell 准则的搜索步长 α_k , Wolfe-Powell 准则描述具体见 2.3 节。

3) 更新搜索构象 \mathbf{x}_{k+1} : $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$, 其中, \mathbf{p}_k 和 α_k 分别是上面计算的搜索方向和步长。

4) 更新海森矩阵 \mathbf{B}_{k+1} :

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{\mathbf{B}_k s_k s_k^T \mathbf{B}_k^T}{s_k^T \mathbf{B}_k s_k} \quad (6)$$

其中, $s_k = \alpha_k \mathbf{p}_k$, $y_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k)$, 海森矩

阵更新需要耗费计算资源,减少 BFGS 迭代搜索的深度可以有效的减少海森矩阵的更新,有效提高计算效率。如果 $\|\nabla f(\mathbf{x}_{k+1})\| \leq \varepsilon$ 或迭代次数 $k = T$, ε 为预定值,则 BFGS 迭代结束,并设局部搜索的新构象 $\bar{C}_i = \mathbf{x}_{k+1}$ 。

Step 4 对于局部搜索得到的新构象 \bar{C}_i ,比较新构象 \bar{C}_i 的能量和扰动前的构象 C_i 的能量,利用模拟退火准则以概率判断是否接受新构象 \bar{C}_i ,如果 \bar{C}_i 被接受继续执行,如果不被接受就返回 Step1,这样可以在一定概率上使得搜索空间跳出局部极小值,收敛到全局最小值,模拟退火准则的接受概率 P 为

$$P = \begin{cases} 1, & E_{C_i} > E_{\bar{C}_i}, \\ \frac{\text{Exp}(E_{C_i} - E_{\bar{C}_i})}{1.2}, & E_{C_i} \leq E_{\bar{C}_i}, \end{cases} \quad (7)$$

其中, $E_{\bar{C}_i}$ 表示新构象 \bar{C}_i 的能量, E_{C_i} 表示扰动前的构象 C_i 的能量,如果 $E_{\bar{C}_i} < E_{C_i}$ 则接受新构象 \bar{C}_i ,如果 $E_{\bar{C}_i} \geq E_{C_i}$,以 $\exp(E_{C_i} - E_{\bar{C}_i})/1.2$ 的概率接受新构象 \bar{C}_i ,通过接受更差的构象可以更广泛地搜索全局最优构象,避免陷入局部最优解。

Step 5 对构象 \bar{C}_i 再进行一次启发式条件判断和局部搜索,选取能量记录更优构象 C_i^* ,如果迭代次数未达到搜索深度 D ,返回 Step 1 对分子构象继续随机扰动,如果迭代次数达到 D ,则输出该线程中的最佳分子构象 C_i^* 。

2.3 基于 Wolfe-Powell 准则的局部搜索算法

蒙特卡罗搜索算法中的局部搜索算法是找到最佳配体分子构象的重要模块,相对于其他模块耗费时间较长。QVina2 利用基于不精确线性搜索 Armijo 准则局部搜索对分子构象进行优化,Armijo 准则的主要思想是确定搜索方向 \mathbf{p}_k ,找到合适的步长 α_k ,保证目标函数值下降,且下降量满足不等式关系:

$$f(\mathbf{x}_k + \alpha_k \mathbf{p}_k) \leq f(\mathbf{x}_k) + c_1 \alpha_k \mathbf{p}_k^T \nabla f(\mathbf{x}_k) \quad (8)$$

其中, c_1 为常数且满足 $c_1 \in (0,1)$ 。在步长更新的过程中,公式(8)主要目的是限制步长,确保下一个搜索构象的能量有足够的下降,但是该条件不能确保步长值接近最佳步长,因为只要步长足够小就可以满足公式(8),如果步长过小会增加 BFGS 搜索迭代的次数。为了克服 Armijo 准则这一缺陷,我们提出使用基于 Wolfe-Powell 准则的局部搜索算法。Wolfe-Powell 属于不精确线性搜索准则,相比于 Armijo 准则,Wolfe-Powell 准则更适合 BFGS 算法,可以保证海森矩阵迭代的正定性,增加了对步长的限制条件:

$$-\mathbf{p}_k^T \nabla f(\mathbf{x}_k + \alpha_k \mathbf{p}_k) \leq -c_2 \mathbf{p}_k^T \nabla f(\mathbf{x}_k) \quad (9)$$

其中, c_2 为常数且满足 $c_2 \in (0,1)$ 。公式(9)的作用是限制过小的步长,保证能量函数的方向导数充分下降,找到更合适的步长,从而以更小的迭代次数找到最佳的分子构象,提高对接精度。因而,QVina2-GPU 通过基于 Wolfe-Powell 准则局部搜索可以进一步减少蒙特卡罗迭代算法的搜索深度,在保证全局优化搜索结果一致的情况下,提高软件对接速度。Wolfe-Powell 准则确定步长的迭代过程如下:

Step 1 给定 $c_1 \in (0,0.5)$, $c_2 \in (c_1,1)$,令 $\alpha_0 = 1$, $n = 0$,设定步长迭代次数 L 。

Step 2 $f(\mathbf{x}_{k+1}) = f(\mathbf{x}_k + \alpha_n \mathbf{p}_k)$,其中 \mathbf{p}_k 和 \mathbf{x}_k 分别是已知的搜索方向和搜索构象, α_n 为第 n 次迭代的步长。

Step 3 若 α_n 满足公式(8)和(9)或步长迭代次数 $n = L$,则 $\alpha_k = \alpha_n$ 并退出步长更新迭代。若 α_n 不满足公式(8),令 $b = \alpha_n$, $\alpha_n = \alpha_n/2$, $n = n + 1$,返回 Step 3。

Step 4 若 α_n 不满足公式(9),令 $\alpha_n = \min(2\alpha_n, 0.5 * (\alpha_n + b))$, $n = n + 1$,返回 Step 3。

综上所述,基于 Wolfe-Powell 准则的蒙特卡罗搜索算法如下面所述,算法流程大体可以分为 5 步,包括随机扰动、启发式条件、局部搜索和模拟退火准则四个模块。基于 Wolfe-Powell 准则的蒙特卡罗搜索算法交给 GPU 的子任务执行,这样大量的子任务可以并行执行搜索算法,数量众多的搜索线程可以实现对分子构象空间的有效搜索,可以减少单独线程蒙特卡罗搜索算法的搜索深度,在保证全局优化搜索结果一致的情况下,显著减少了分子对接软件运行时间。

3 实验及性能分析

3.1 实验环境及数据集

QVina2-GPU 的异构并行架构通过开放运算语言 OpenCL 实现,实验环境的 CPU 为 4 核的 Intel (R) Xeon(R) Gold 6130 CPU @ 2.10GHz, GPU 为 NVIDIA GeForce RTX 3090,操作系统为 Windows 10,编译器为 Visual Studio 2019, CUDA 的版本为 11.1, OpenCL 的版本为 3.0。

图 3 显示了配体 MGT1484 和药物靶标受体 PROTEIN 分子对接结果的 3D 结构示意图,图中左侧框图为分子对接过程的对接口袋,右侧框图为具体的对接过程,受体和配体之间的虚线为分子作用力。分子对接软件就是通过打分函数计算分子对接过程的结合能量,来预测药物靶标受体和配体结合

的可能性,选出具有可能成药的活性化合物。

算法:基于 Wolfe-Powell 准则的蒙特卡罗搜索算法

输入:

- ① N 个随机初始构象 $\{C_1, \dots, C_N\}$;
- ② 搜索深度: D ;
- ③ set $d = 0$;
- ④ Parallel for C_i ($i = 0, \dots, N$) do
- ⑤ while $d < D$ do
- ⑥ 根据公式(2)随机扰动 C_i 得到候选构象 \bar{C}_i ;
- ⑦ If \bar{C}_i 满足公式(4a)或(4b)的启发式条件
- ⑧ 根据基于 Wolfe-Powell 准则的 BFGS 搜索算法 得到新构象 \bar{C}_i ;
- ⑨ 根据公式(7)计算模拟退火算法接受概率 P ;
- ⑩ If 以 P 概率接受 \bar{C}_i & \bar{C}_i 满足启发式条件
- ⑪ 根据基于 Wolfe-Powell 准则的 BFGS 搜索算法 得到新构象 C_i^*
- ⑫ end if
- ⑬ end if
- ⑭ $d = d + 1$;
- ⑮ end while

输出:排名前 k 个最佳分子构象 $\{C_0^*, C_1^*, \dots, C_{k-1}^*\}$

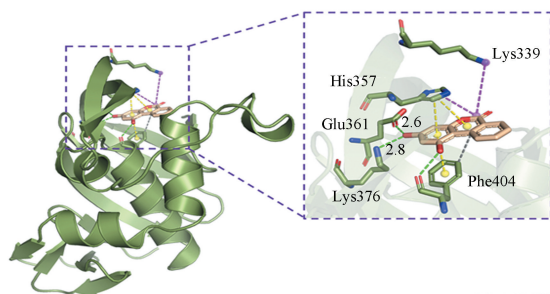


图3 药物靶标受体和配体分子对接示意图

Fig.3 Schematic diagram of drug target receptor and ligand molecular docking

使用包含 140 个药物靶标受体-配体复合物的数据集作为实验数据集,其中 85 个来自 Astex 多样性集^[23]、35 个来自 CASF-2013^[24]和 20 个来自蛋白质数据库^[25],该数据集包含了广泛的配体复杂度和靶标性质。数据集根据原子数目 N_{atom} 分为简单复合物、中等复杂复合物和复杂复合物,其中,简单复合物的数量为 46,原子数目 N_{atom} 范围为 $[5, 23]$,中等复杂复合物的数量为 51, N_{atom} 范围为 $[24, 36]$,复杂复合物的数量为 43, N_{atom} 范围为 $[37, 108]$,三种复合物对应配体复杂度越来越高。

3.2 实验指标和对比算法

对接精度和对接时间是评估分子对接软件性能最重要的两个指标,对接精度主要由对接分数 Score 和 RMSD(Root mean square deviation)决定,1)对接分数 Score 表示配体和受体之间的结合能,Score 越低则结合越稳定,2)RMSD 表示分子对接软件搜索得到的构象与生物实验获得的 X-ray 结构(标准)之间的相似性,RMSD 越低则表示搜索到的构象精度越高,当 RMSD 小于 2\AA (10^{-10} m)时,该构象是

可以被接受,表示成功预测。3)对接时间是整个分子对接软件的运行时间,对接时间越短,对接精度越高,表示分子对接软件性能越好。

在相同的实验环境中比较 QVina2 与 QVina2-GPU 分子对接软件的性能,两种算法的介绍如下:

1) QVina2(Quick Vina2):是一种快速准确的分子对接软件,并提出了启发式算法改进了局部搜索算法,与传统的分子对接软件相比提高了运行速度。

2) QVina2-GPU:是我们在 QVina2 对接软件基础上,提出的一种异构并行架构并实现了基于 Wolfe-Powell 准则的蒙特卡罗搜索,利用 GPU 硬件高度并行体系加速分子对接过程。

在分子对接软件运行前,需要先设置配置文件。QVina2 的配置文件中的配置参数包括搜索空间的中心、对接盒子的体积,另外还包括两个重要的参数:初始化分子构象数量 N 和搜索深度 D ,二者反应的分子构象搜索的广度和深度,其中分子对接软件的运行时间主要由搜索深度 D 决定,

3) QVina2 默认参数 N 为 8,搜索深度 D 随着配体分子的复杂度增加而增大,对于简单分子复合物、中等复杂复合物和复杂复合物的平均搜索深度分别为 16 170、29 663 和 41 212。

QVina2-GPU 利用 GPU 并行化处理能力大量增加初始化分子构象数量 N ,使得大量线程分别对应不同的分子初始构象,使得每个线程独立承担蒙特卡罗搜索算法子任务,独立搜索最佳的分子构象。以减少每个子任务的每次搜索深度 D ,在保证全局优化搜索结果一致的情况下,显著减少了分子对接软件运行时间。下面将重点讨论初始化分子构象数量 N 和搜索深度 D 对性能的影响。

3.3 两种准则对接精度比较

比较对象 QVina2 使用的是基于 Armijo 准则的蒙特卡罗搜索算法,我们提出的 QVina2-GPU 使用的是基于 Wolfe-Powell 准则的蒙特卡罗搜索算法。因而,本次实验讨论基于 Wolfe-Powell 准则的蒙特卡罗搜索算法和基于 Armijo 准则的蒙特卡罗搜索算法的对接精度,即对接分数 Score 和 RMSD 指标。

在 140 个配体小分子复合物上进行对接实验,对比两种准则下对接分数 Score 和 RMSD 的影响,实验结果如图 4 所示,其中图 4a 表示 140 个配体小分子在两种准则条件下的 Score,其中 Score 的单位是 kcal/mol,表示每摩尔分子千卡能量,图 4b 表示两种准则条件下的 RMSD 指标。两幅图的纵坐标表示基于 Armijo 准则的指标,横坐标表示基于 Wolfe-Powell 准则的指标。颜色条表示配体中的原

子数,颜色条从深到浅对应原子数目由少至多,即配

体分子复杂度越来越高。

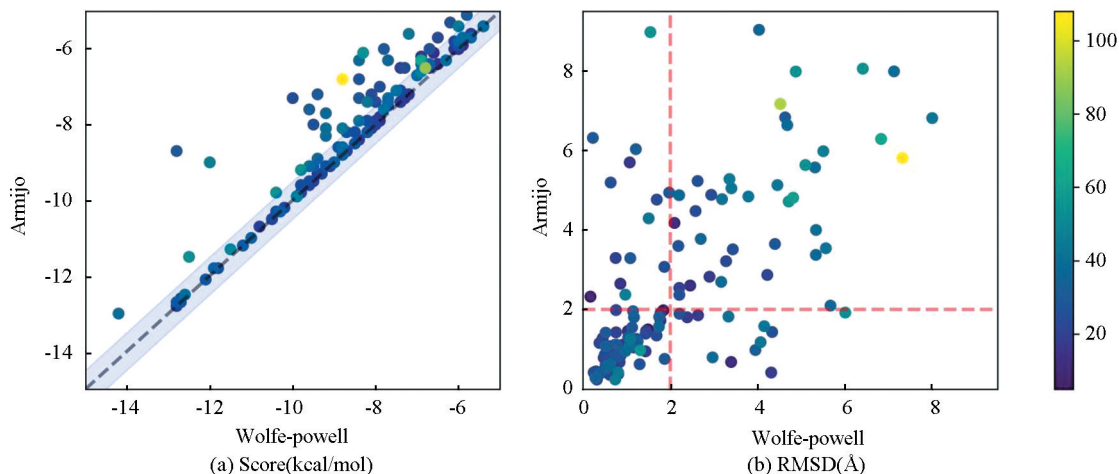


图 4 两种准则的 Score 和 RMSD 对比图 (Score 和 RMSD 越小越好)

Fig.4 Comparison diagram of score and RMSD of the two criteria (the smaller the score and RMSD, the better)

结果显示数据集中 60.71% 的复合物位于图 4a 的对角线上方,表明使用 Wolfe-Powell 准则得到的对接分数 Score 小于 Armijo 准则的 Score,即 Wolfe-Powell 准则性能优于 Armijo 准则;有 39.29% 的复合物位于图 4a 的对角线上,表明用 Wolfe-Powell 准则得到的 Score 与 Armijo 准则相同;基于 Wolfe-Powell 准则的蒙特卡罗搜索算法有 63.57% 构象位于图 4b 中垂直红色虚线左边区域内,表示 RMSD 指标小于 2\AA (10^{-10} m) 可被接受,而基于 Armijo 准则蒙特卡罗搜索算法只有 59.28% 构象位于图 4b 水平红色虚线下方区域内,可被接受。结果表明,相对于 QVina2 中基于 Armijo 准则蒙特卡罗搜索算法,我们提出的基于 Wolfe-Powell 准则的蒙特卡罗搜索算法具有更高的对接精度,在 Score 和 RMSD 指标中均明显优于基于 Armijo 准则蒙特卡罗搜索算法。

3.4 QVina2-GPU 中初始化分子构象数和搜索深度对性能的影响

不失一般性,从实验数据集中随机选取 8 个复合物, QVina2 的初始化分子构象数 N 和搜索深度 D

按照配置文件设置并固定不变, QVina2-GPU 的搜索深度 D 设置为 1, 初始化分子构象数 N 从 100 增大到 8 000, 讨论 N 变化对 QVina2-GPU 的对接分数 Score, RMSD 和对接时间的影响。

如表 1-表 3 所示,复合物从上到下对应配体复杂度越来越高,表 1-表 2 可以看出初始化分子构象数 N 为 800 可以保证 QVina2-GPU 的 Score 和 RMSD 指标与 QVina2 相当,继续增加初始化分子构象数 N , QVina2-GPU 的 Score 和 RMSD 指标将优于 QVina2。表 3 的实验结果表明在所有情况下 QVina2-GPU 对接时间相对于 QVina2 有了较大的降低,而且 QVina2-GPU 对接时间随着初始化分子构象数 N 的增加而增加,对于复杂复合物这种增加的趋势更加明显,因此,可以得出:增加初始化分子构象数 N , 可以有效的提高分子对接精度,但同时也增加了对接时间开销。综上所述,为了在保证精度前提下尽量减少分子对接时间,下面的实验中,将 QVina2-GPU 的初始化分子构象数 N 设置为 800。

表 1 初始化分子构象数 N 对 Score 的影响 (Score 越小越好, 标粗表示性能达到或优于 QVina2)

Table 1 Effect of initial molecular conformation number N on score

(The smaller the score, the better. The bold numbers mean that the performance reaches or exceeds QVina2)

分子类型	Score/ kcal · mol ⁻¹								QVina2
	QVina2-GPU								
	100	300	500	800	1 000	3 000	5 000	8 000	
5tim	-3.70	-3.70	-3.80	-3.80	-3.80	-3.80	-3.80	-3.80	-3.80
1aet	-3.90	-3.90	-4.0	-4.0	-4.0	-4.0	-4.0	-4.0	-4.0
1ac8	-3.90	-3.90	-3.90	-3.90	-3.90	-3.90	-3.90	-3.90	-3.90
1hvy	-5.30	-8.50	-8.20	-8.80	-8.80	-10.00	-9.30	-9.40	-8.80
1os0	-5.90	-6.00	-7.10	-7.80	-7.60	-7.50	-8.70	-8.00	-6.30
2er7	-4.30	-4.40	-5.10	-6.20	-6.50	-6.00	-6.60	-6.90	-4.60
4er4	-5.30	-5.70	-5.40	-6.60	-6.80	-7.20	-6.60	-7.60	-5.80
3er5	-3.60	-4.40	-4.50	-6.50	-7.10	-6.50	-7.00	-7.10	-6.30

表 2 初始化分子构象数 N 对于 RMSD 的影响 (RMSD 越小越好,标粗表示性能达到或优于 QVina2)Table 2 Effect of initial molecular conformation number N on RMSD

(The smaller the RMSD, the better. The bold numbers indicate that the performance reaches or exceeds QVina2)

分子类型	RMSD/Å								QVina2
	QVina2-GPU								
	100	300	500	800	1 000	3 000	5 000	8 000	
5tim	2.11	1.71	1.99	1.48	1.91	1.84	1.84	1.93	1.98
1aet	1.73	1.49	1.74	1.49	1.49	1.49	1.49	1.49	1.50
1ac8	2.50	2.48	2.49	2.33	2.45	2.49	2.49	0.15	2.69
1hvy	7.76	1.60	1.78	1.36	1.21	1.27	1.18	1.50	2.19
1os0	3.07	1.61	1.61	2.36	2.24	1.34	4.28	1.41	6.64
2er7	13.65	6.35	8.67	5.81	8.33	5.10	5.92	5.49	9.89
4er4	10.98	8.53	4.85	5.95	5.88	5.27	5.26	6.79	7.17
3er5	18.72	7.66	8.67	7.58	6.49	6.62	6.47	6.85	8.12

表 3 初始化分子构象数 N 对于对接时间的影响 (对接时间越小越好,标粗表示性能达到或优于 QVina2)Table 3 Effect of initial molecular conformation number N on docking time

(The smaller the docking time, the better. The bold numbers indicate that the performance reaches or exceeds QVina2)

分子类型	Run time/s								QVina2
	QVina2-GPU								
	100	300	500	800	1 000	3 000	5 000	8 000	
5tim	1.97	1.97	1.94	1.90	1.98	2.18	2.15	2.38	4.11
1aet	2.55	2.56	2.47	2.52	2.57	2.74	2.68	2.81	4.69
1ac8	3.04	2.96	3.10	3.07	3.25	3.46	3.24	3.61	4.38
1hvy	7.76	3.80	3.80	3.97	4.38	4.12	4.33	4.57	7.43
1os0	3.67	3.08	3.29	3.21	3.31	3.29	3.43	3.78	18.12
2er7	3.09	5.93	5.95	6.14	7.55	7.38	7.53	8.95	24.67
4er4	6.40	5.70	5.73	5.88	6.01	6.80	7.30	7.66	32.80
3er5	5.83	6.91	7.32	7.75	7.80	8.71	9.16	9.43	25.13

在确定初始化分子构象数 N 后,我们将搜索深度 D 从 1 增大到 30,探究 D 对于 QVina2-GPU 的对接精度和对接时间的影响,表 4-表 6 可以看出对于简单复合物如 5tim 等,增大搜索深度 D 对 Score、RMSE 和对接时间的影响很小;对于中等复杂复合物如 1hvy 等,随着搜索深度 D 增加,对接精度指标 Score 和 RMSE 总体上有所减少,但对接时间缓慢增加;对于复杂复合物 3er5 等,增大搜索深度 D ,对接

精度指标 Score 和 RMSE 同样总体上有所减少,但对接时间增加很快,会导致很大的时间开销。

综上所述,为了保证精度前提下尽量减少分子对接时间,特别是考虑到搜索深度 D 对于复杂复合物对接时间的影响特别明显。因此,接下来的实验,我们将数据集 140 个复合物的初始化分子构象数 N 设为 800,搜索深度 D 设为 1,以最少的对接时间保证了分子对接精度。

表 4 搜索深度 D 对 Score 的影响 (Score 越小越好,标粗表示性能达到或优于 QVina2)Table 4 Effect of search depth D on score

(The smaller the score, the better. The bold numbers indicate that the performance reaches or exceeds QVina2)

分子类型	Score/ kcal · mol ⁻¹							QVina2
	QVina2-GPU							
	1	2	3	5	10	15	30	
5tim	-3.80	-3.80	-3.80	-3.80	-3.80	-3.80	-3.80	-3.80
1aet	-4.0	-4.00	-4.00	-4.00	-4.00	-4.00	-4.00	-4.00
1ac8	-3.90	-3.90	-3.90	-3.90	-3.90	-3.90	-3.90	-3.90
1hvy	-8.80	-9.60	-9.30	-9.60	-9.20	-9.50	-9.30	-8.80
1os0	-7.80	-7.90	-8.40	-8.30	-8.20	-9.00	-8.60	-6.30
2er7	-6.20	-6.90	-5.90	-6.30	-6.10	-7.10	-8.00	-4.60
4er4	-6.60	-6.60	-6.90	-7.10	-6.90	-6.90	-7.00	-5.80
3er5	-6.50	-6.10	-6.00	-6.40	-6.80	-7.90	-7.60	-6.30

表 5 搜索深度 D 对 RMSD 的影响 (RMSD 越小越好, 标粗表示性能达到或优于 QVina2)Table 5 Effect of search depth D on RMSD

(The smaller the RMSD, the better. The bold numbers indicate that the performance reaches or exceeds QVina2)

分子类型	RMSD/Å							QVina2
	QVina2-GPU							
	1	2	3	5	10	15	30	
5tim	1.48	1.28	1.84	1.58	2.08	2.08	1.48	1.98
1aet	1.49	1.49	1.49	1.49	1.49	1.49	1.50	1.50
1ac8	2.33	2.49	2.49	2.52	2.49	2.49	2.49	2.69
1hvy	1.36	1.49	1.18	0.97	0.95	0.79	1.33	2.19
1os0	2.36	2.26	2.00	4.88	1.67	1.36	1.34	6.64
2er7	5.81	6.27	5.89	5.72	6.33	6.18	5.59	9.89
4er4	5.95	7.32	4.88	6.54	8.16	4.78	4.88	7.17
3er5	7.58	7.87	6.59	6.17	7.45	6.29	6.56	8.12

表 6 搜索深度 D 对于对接时间的影响 (对接时间越小越好, 标粗表示性能达到或优于 QVina2)Table 6 Effect of Search Depth D on docking time

(The smaller the docking time, the better. The bold numbers indicate that the performance reaches or exceeds QVina2)

分子类型	Run time/s							QVina2
	QVina2-GPU							
	1	2	3	5	10	15	30	
5tim	1.90	2.09	1.99	2.11	2.10	2.09	1.99	4.113
1aet	2.52	2.53	2.47	3.00	2.57	2.43	2.47	4.69
1ac8	3.07	3.17	3.16	3.59	3.82	3.12	3.17	4.38
1hvy	3.97	4.09	4.22	5.76	4.61	4.61	4.69	7.43
1os0	3.21	3.48	3.80	4.38	4.61	4.86	4.62	18.12
2er7	6.14	9.02	12.24	17.69	23.11	22.87	23.24	24.67
4er4	5.88	8.69	11.68	17.08	21.33	20.86	20.83	32.80
3er5	7.75	11.21	15.29	23.27	28.30	27.92	27.84	25.13

3.5 QVina2-GPU 与 QVina2 的性能对比

首先对比了 QVina2-GPU 和 QVina2 在 140 个复合物上的对接分数 Score 和 RMSD 对接精度指标, 两幅图的纵坐标表示 QVina2 的性能指标, 横坐标表示 QVina2-GPU 的性能指标。颜色条表示配体中的原子数, 颜色条从深到浅对应原子数目由少至多, 即配体分子复杂度越来越高。

对接分数 Score 结果如图 5a 所示, 大多数复合物分布在对角线周围, 其对接分数的皮尔森相关系数为 0.914, 表示它们之间存在极强相关性, 即 QVina2-GPU 和 QVina2 对接分子 Score 的指标基本相同; 图 5b 显示大多数复合物落入左下角区域, QVina2 (水平红色虚线下方区域内) 和 QVina2-GPU (垂直红色虚线下方区域内) 分别有 58.28% 和 55% 的复合物的预测结果可被接受, 结果表明本文提出的 QVina2-GPU 在初始化分子构象数 N 为 800 和搜索深度 D 为 1 的条件下,

达到了 QVina2 的对接精度。

下面重点讨论一下 QVina2-GPU 和 QVina2 在 140 个复合物上的对接时间指标, 这里我们使用两个对接时间指标, 对接时间加速比 Acc 和蒙特卡罗时间加速比 Acc_d , 定义如下:

$$Acc = \frac{T_{\text{qvina2}}}{T_{\text{qvina2-gpu}}}, Acc_d = \frac{T_{\text{mc}}}{T_{\text{mc-gpu}}} \quad (10)$$

式中, T_{qvina2} 是 QVina2 分子对接计算的对接时间, $T_{\text{qvina2-gpu}}$ 是 QVina2-GPU 分子对接计算的对接时间。QVina2 与 QVina2-GPU 软件运行中, 蒙特卡罗迭代搜索算法都是最耗时的部分, 因而, T_{mc} 为 QVina2 中基于 CPU 的蒙特卡罗算法的运行时间, $T_{\text{mc-gpu}}$ 为 QVina2-GPU 的蒙特卡罗运行时间。对接时间加速比 Acc 和蒙特卡罗时间加速比 Acc_d 数值越大, 说明我们提出的 QVina2-GPU 相对于 QVina2 有较高的加速比, 即运行时间相对于 QVina2 有显著的减少。

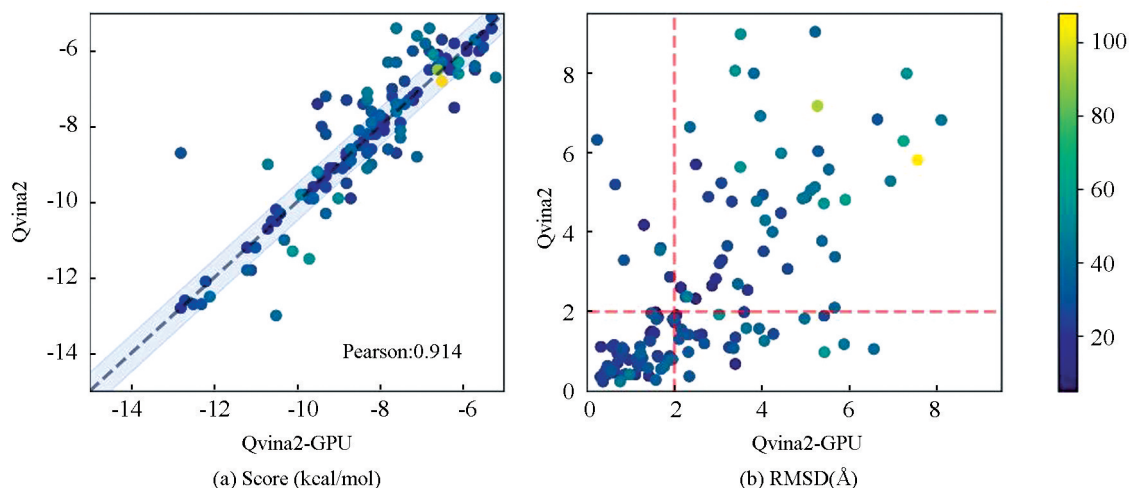


图5 QVina2-GPU与QVina2的Score和RMSD对接精度对比图($N = 800, D = 1$)

Fig. 5 Comparison chart of score and RMSD docking accuracy between QVina2-GPU and QVina2 ($N = 800, D = 1$)

图6和图7给出了140个复合物的对接时间加速比 Acc 和蒙特卡罗时间加速比 Acc_d 。为了区分不同复杂度配体分子的加速比情况,三维坐标的横纵坐标分别表示配体的原子数目 N_{atom} 和可旋转键数目 N_{rot} ,配体的复杂度由原子数目 N_{atom} 和可旋转键数目 N_{rot} 决定,一般来说 N_{atom} 和 N_{rot} 越大,配体的复杂度越高。图6和图7中每个条形柱体分别表示配体的复杂程度和对应的对接时间加速比 Acc 和蒙特卡罗时间加速比 Acc_d ,对接时间加速比的蓝色柱状条表示加速比达到10的复合物,蒙特卡罗时间加速比的蓝色柱状条为加速比达到60的复合物,越靠近蓝色加速比越高,图5可以看出 Acc 分布范围在3.11倍~12.28倍之间,图7可以看出 Acc_d 分布范围为6.48倍~60.28倍,其中,2bm2复合物加速效果最好,它的蒙特卡罗时间加速比 Acc 达到了60.28倍,对接时间加速比 Acc_d 达到了12.28倍。上面数据充分说明了我們提出的QVina2-GPU在不同类型的复合物上都有明显的加速效果,显著的减少了分子对接的计算时间,这对于需要大量筛选配体分子的生物实验特别有意义。

为进一步显示QVina2-GPU加速比性能,三种配体分子复杂度下的平均加速比结果(见表7),结果表明,最大平均蒙特卡罗时间加速比为22.91倍,最大平均对接时间加速比为5.71倍,三种不同复杂度复合物条件下,QVina2-GPU都取得了令人满意的加速比,总体平均蒙特卡罗时间加速比和平均对接时间加速比分别到达了18.63和5.18。从上面的分析可知,QVina2-GPU利用GPU并行化处理能力大量增加初始化分子构象数量 N ,使得大量线程分别对应不同的分子初始构象,使得每个线程独立承担

蒙特卡罗搜索算法子任务,增加初始化分子构象数量 N 以减少每个子任务的每次搜索深度 D ,从而显著减少了蒙特卡罗构象搜索时间,减少了分子对接软件的整体运行时间。平均对接时间加速比随着复杂度的增大而提高,这是由于随着复杂度的增加,蒙特卡罗搜索算法运行时间在整个软件运行时间比例加重,平均对接时间加速比随着增加。

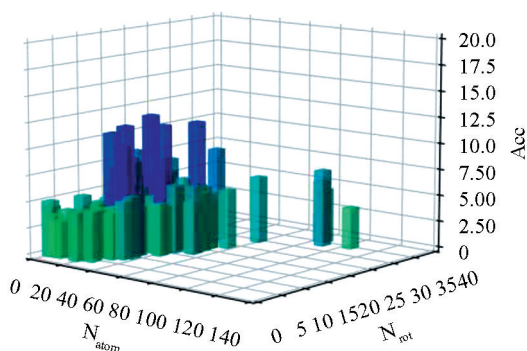


图6 140个复合物的对接时间加速比

Fig.6 Docking time acceleration ratio of 140 composites

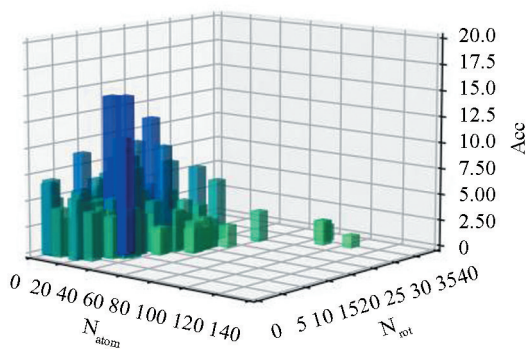


图7 140个复合物的蒙特卡罗时间加速比

Fig.7 Monte Carlo time acceleration ratio of 140 compounds

表 7 三种配体复杂度的平均时间和平均时间加速比

Table 7 Average time and average time acceleration ratio of three ligand complexities

分子类型	QVina2		QVina2-GPU		平均蒙特卡罗 时间加速比	平均对接 时间加速比
	平均蒙特卡罗 时间/s	平均对接 时间/s	平均蒙特卡罗 时间/s	平均对接 时间/s		
简单复合物	2.55	5.04	0.13	1.24	19.00	4.14
中等复杂复合物	4.98	7.69	0.21	1.38	22.91	5.66
复杂复合物	8.52	10.94	0.82	1.95	13.18	5.71
总体平均值	5.27	7.82	0.37	1.51	18.63	5.18

3.6 QVina2-GPU 与 Vina-GPU 性能对比

为了区分 Vina-GPU^[26] 和 QVina2-GPU 两种方法在性能上的不同,我们从 Vina-GPU 论文里获取可执行程序,在相同的硬件平台实验环境下,对上文提到的随机 8 个复合物对比 QVina2-GPU 与 Vina-GPU 的对接分数 Score 和 RMSD 对接精度指标和对接时

间,如表 8 所示,实验表明,对于简单分子和中等复杂分子 QVina2-GPU 可以在对接精度基本相同的情况下加速 Vina-GPU,对于复杂分子,我们提出的 QVina2-GPU 在对接精度略微下降的条件下,运行时间上有明显的减少,相对于 Vina-GPU 的加速比最大可达 32.92,这对于实际药物筛选应用具有很大的意义。

表 8 Vina-GPU 与 QVina2-GPU 的性能对比

Table 8 Performance comparison between Vina GPU and QVina2 GPU

分子类型	Vina-GPU			QVina2-GPU			对接时间 加速比
	Score/ (kcal · mol ⁻¹)	Rmsd/Å	Run time/s	Score/ (kcal · mol ⁻¹)	Rmse/Å	Run time/s	
5tim	-3.8	2.08	2.71	-3.8	1.48	1.9	1.43
1aet	-4	1.49	2.98	-4	1.49	2.52	1.18
1ac8	-3.9	2.49	3.25	-3.9	2.33	3.07	1.06
1hvy	-9.6	0.66	5.74	-8.8	1.36	3.97	1.45
1os0	-8.1	1.43	7.42	-7.8	2.36	3.21	2.31
2er7	-8.4	2.22	169.62	-6.2	5.81	6.14	27.63
4er4	-8.5	6.3	165.97	-6.6	5.95	5.88	28.23
3er5	-8.3	6.12	255.10	-6.5	7.58	7.75	32.92

4 结 论

本文提出基于 GPU 加速的分子对接软件并行化方法 QVina2-GPU,针对 Qvina2 在大型虚拟数据库中筛选时间过长的情况,通过增加初始化分子构象数量来扩展蒙特卡罗的迭代局部搜索中线程的并行规模,减少蒙特卡罗迭代搜索算法的搜索步数,而且利用 Wolfe-Powell 准则改进局部搜索算法,提高了对接精度,进一步减少蒙特卡罗迭代算法的搜索深度,提高分子对接软件的运行速度。在公开的药物靶标受体-配体复合物的数据集上验证了不同类型复合物上的平均蒙特卡罗时间加速比和平均对接时间加速比,结果表明 QVina2-GPU 在达到 Qvina2 分子对接精度的条件下,相对于 QVina2 有显著的加速效果。QVina2-GPU 的代码可以在 <https://github.com/DeltaGroupNJUPT/QuickVina2-GPU> 上获得,其中包含可执行程序 and 程序使用说明,便于科研人员使用。

由于 QVina2-GPU 计算构象的能量使用的依旧

是 QVina2 的能量计算函数,是一种经验公式,筛选出能量较低的复合物用于生物实验中,存在较高的偏差^[27],近年来已有学者研究基于深度学习的能量计算函数。未来,我们的工作将通过研究新的能量计算函数来提高 QVina2-GPU 的对接性能。

参考文献(References)

- [1] KONTTOYIANNI M. Docking and virtual screening in drug discovery[J]. *Methods in Molecular Biology*, 2017, 1647: 255-266. DOI: 10.1007/978-1-4939-7201-2_18.
- [2] WALTERS W P, STAHL M T, MURCKO M A. Virtual screening—an overview[J]. *Drug Discovery Today*, 1998, 3(4): 160-178. DOI: 10.1016/S1359-6446(97)01163-X.
- [3] MURCKO M, CARON P. Transforming the genome to drug discovery[J]. *Drug Discovery Today*, 2002, 7(11): 583-584. DOI: 10.1016/S1359-6446(02)02267-5.
- [4] LAVECCHIA A, GIOVANNI C. Virtual screening strategies in drug discovery: a critical review[J]. *Current Medicinal Chemistry*, 2013, 20(23): 2839-2360. DOI: 10.2174/

- 09298673113209990001.
- [5] KOOISTRA A J, VISCHER H F, MCNAUGHT-FLORES D, et al. Function-specific virtual screening for GPCR ligands using a combined scoring method [J]. *Scientific Reports*, 2016, 6: 28288. DOI: 10.1038/srep28288.
- [6] BALLESTER P J. Selecting machine-learning scoring functions for structure-based virtual screening [J]. *Drug Discovery Today: Technologies*, 2019, 32–33: 81–87. DOI: 10.1016/j.ddtec.2020.09.001.
- [7] LI Hongjian, SZE K H, LU Gang, et al. Machine-learning scoring functions for structure-based virtual screening [J]. *Wiley Interdisciplinary Reviews-Computational Molecular Science*, 2021; 11(1): e1478. DOI: 10.1002/wcms.1428.
- [8] GEPPERT H, VOGT M, BAJORATH J. Current trends in ligand-based virtual screening: molecular representations, data mining methods, new application areas, and performance evaluation [J]. *Journal of Chemical Information and Modeling*, 2010, 50(2): 205–216. DOI: 10.1021/ci900419k.
- [9] SANTANA K, DO NASCIMENTO L D, LIMA E, LIMA A, et al. Applications of virtual screening in bioprospecting: facts, shifts, and perspectives to explore the chemo-structural diversity of natural products [J]. *Frontiers in Chemistry*, 2021, 9: 662688. DOI: 10.3389/fchem.2021.662688.
- [10] KUBINYI H. Similarity and dissimilarity: A medicinal chemist's view [J]. *Perspect in Drug Discovery and Design*, 1998(9/10/11): 225–252. DOI: 10.1023/A:1027221424359.
- [11] MORRIS G M, HUEY R, LINDSTROM W, et al. AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility [J]. *Journal of Computational Chemistry*. 2009, 30(16): 2785–2791. DOI: 10.1002/jcc.21256.
- [12] TROTT O, OLSON A J. AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading [J]. *Journal of Computational Chemistry*. 2009, 31(2): 455–461. DOI: 10.1002/jcc.21334.
- [13] EBERHARDT J, SANTOS-MARTINS D, TILLACK A, et al. AutoDock Vina 1.2.0: New docking methods, expanded force field, and python bindings [J]. *Journal of Chemical Information And Modeling*, 2021, 61(8): 3891–3898. DOI: 10.26434/chemrxiv.14774223.
- [14] KOES D R, BAUMGARTNER M P, CAMACHO C J. Lessons learned in empirical scoring with smina from the CSAR 2011 benchmarking exercise [J]. *Journal of Chemical Information and Modeling*, 2013, 53(8): 1893–1904. DOI: 10.1021/ci300604z.
- [15] GORGULLA C, BOESZOERMENYI A, WANG Z F, et al. An open-source drug discovery platform enables ultra-large virtual screens [J]. *Nature*, 2020, 580(7805): 663–668. DOI: 10.1038/s41586-020-2117-z.
- [16] HANDOKO S D, OUYANG X, SU C T, et al. QuickVina: Accelerating AutoDock Vina using gradient-based heuristics for global optimization. [J]. *IEEE/ACM Transactions on Computational Biology & Bioinformatics*, 2012, 9(5): 1266–1272. DOI: 10.1109/TCBB.2012.82.
- [17] ALHOSSARY A, HANDOKO S D, MU Y, et al. Fast, Accurate, and reliable molecular docking with quickVina 2 [J]. *Bioinformatics*, 2015, 31(13): 2214–2216. DOI: 10.1093/bioinformatics/btv082.
- [18] DING Xinqiang, WU Yujin, WANG Yanming, et al. Accelerated cdocker with gpus, parallel simulated annealing, and fast fourier transforms [J]. *Journal of Chemical Theory and Computation*, 2020, 16(6): 3910–3919. DOI: 10.1021/acs.jctc.0c00145.
- [19] SOLIS-VASQUEZ L, SANTOS-MARTINS D, TILLACK A F, et al. Parallelizing irregular computations for molecular docking [C]. // In 2020 IEEE/ACM 10th Workshop on Irregular Applications: Architectures and Algorithms (IA3). 2020: 12–21.
- [20] FAN Mengran, WANG Jian, JIANG Huaipan, et al. Gpu-accelerated flexible molecular docking [J]. *The Journal of Physical Chemistry B*, 2021, 125(4): 1049–1060. DOI: 10.1021/acs.jpcc.0c09051.
- [21] LEGRAND S, SCHEINBERG A, TILLACK A F, et al. Gpu-accelerated drug discovery with docking on the summit supercomputer: porting, optimization, and application to covid-19 research [C]. // Proceedings of the 11th ACM international conference on bioinformatics, computational biology and health informatics. 2020: 1–10. DOI: 10.1145/3388440.3412472.
- [22] SANTOS-MARTINS D, SOLIS-VASQUEZ L, TILLACK A F, et al. Accelerating AutoDock4 with GPUs and gradient-based local search [J]. *Journal of Chemical Theory and Computation*, 2021, 17(2): 1060–1073. DOI: 10.1021/acs.jctc.0c01006.
- [23] HARTSHORN M J, VERDONK M L, CHESSARI G, et al. Diverse, high-quality test set for the validation of proteinligand docking performance [J]. *Journal of Medicinal Chemistry*, 2007, 50(4): 726–741. DOI: 10.1021/jm061277y.
- [24] LI Yan, SU Minyi, LIU Zhihai, et al. Assessing proteinligand interaction scoring functions with the CASF-2013 benchmark [J]. *Nature Protocols*, 2018, 13(4): 666. DOI: 10.1038/nprot.2017.114.
- [25] BERMAN H M, BATTISTUZZI T, BHAT T N, et al. The protein data bank [J]. *Nucleic Acids Research*, 2000, 28(1): 235–242. DOI: 10.1093/nar/28.1.235.
- [26] TANG Shidi, CHEN Ruiqi, LIN Mengru, et al. Accelerating AutoDock Vina with GPUs [J]. *Molecules*, 2022, 27(9): 3041. DOI: 10.3390/molecules27093041.
- [27] SU Minyi, YANG Qifan, DU Yu, et al. Comparative assessment of scoring functions: The CASF-2016 Update [J]. *Journal of Chemical Information And Modeling* 2019; 59: 895–913. DOI: 10.1021/acs.jcim.8b00545.