БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ BULGARIAN ACADEMY OF SCIENCES

ПРОБЛЕМИ НА ТЕХНИЧЕСКАТА КИБЕРНЕТИКА И РОБОТИКАТА, **68**

PROBLEMS OF ENGINEERING CYBERNETICS AND ROBOTICS, **68**

София ● 2017 ● Sofia

Application Of GPU In The Field Of Bioisosterism

Desislava Y. Ivanova

Institute of Information and Communication Technologies Bulgarian Academy of Sciences,
Acad. Georgi Bonchev Str. Block 2, 1113 Sofia, Bulgaria
Email: ***divanovaacomin@gmail.com***

Abstract*: Software applications for 3D visualization and evaluation of bioisosteric molecules are very important in bioisoterism. The use of GPUs for parallel processing of matrix-based operation - for the calculations of quantum vectors - would lead to repeated acceleration of the speed of these calculations. In this matter GPU is a factor, which is important for the improvement of the evaluation methods of studying and visualization of molecular similarity.*

Key words: *3D visualization, bioisosteric molecules, software applications, innovative methods, bisisosterism, GPUs*

1. Bioisosteres In Drug Discovery

Bioisosterism has a crutial role in the development of drug molecules almost from the origin of the pharmaceutical industry. The aim of bioisosterism is that the properties of a compound can be fine-tuned without affecting its fundamental biological activity. However this comes with its challenges. Successfully applying bioisosterism to achieve the intended molecular outcome is difficult because of the fundamental problem that chemical structure is an unreliable indicator of biological activity. A slight change in a molecule can have a far-reaching impact on a compound’s activity, specificity and toxicity, in the same time completely different chemotypes may have near identical biological activity profiles. More exact and reproducible methods for suggesting relevant, non-obvious and yet synthetically intuitive bioisosteres would have wide applicability.

Bioisosteres are used by researchers throughout the pharmaceutical industry to find new hits and leads by modifying known actives or substrates, to develop leads by modifying physicochemical properties and protecting their knowledge using patents. Having identified an interesting target, researchers often had little choice in finding an active inhibitor or antagonist, except through bioisosteric modification of the natural ligand in a systematic and thoughtful manner. The modern HTS era has provided a lot of potential leads, but still the need for bioisosteres stays actual as structures found through HTS can have undesirable properties (either physical or biological) and often lack novelty.

The requirement to protect research positions through patent applications is crucial for the development of new medicines. In this respect, IP protection is probably the most important use of bioisosteres in the modern drug discovery project. Replacement of core groups in the lead series with new scaffolds that introduce better selectivity or physical properties. Scaffold hopping is a computational technique of replacing portions of molecules to create novel drug-like compounds with similar activity to the original. The method involves choosing a portion of the starting molecule, often the central scaffold, for replacement. The scaffold-hopping software searches a database of hundreds of thousands of fragments for the best replacements. The worth of the software depends on the algorithm used to evaluate the “best” matches. Different software tools use different approaches. Some use simple geometrical considerations and/or the presence of simple pharmacophore points while others use ligand similarity to rank the replacements. In all cases, the best matches are returned as possible candidates for synthesis. Choosing the right compounds to progress is important as it can frequently take up to a week or more of lab time to synthesize a new compound. Results must be imaginative, yet realistic suggestions that enable users to advance the compounds that are most likely to succeed.

High-throughput screening (HTS) is a method for scientific experimentation especially used in drug discovery and relevant to the fields of biology and chemistry. Using robotics, data processing and control software, liquid handling devices, and sensitive detectors, High-throughput screening allows a researcher to quickly conduct millions of chemical or pharmacological tests. Through this process one can rapidly identify active compounds that modulate a particular biomolecular pathway. The results of these experiments provide starting points for drug design and for understanding the interaction or role of a particular biochemical process in biology. It still takes a highly specialized and expensive screening lab to run an HTS operation, so in many cases a small- to moderate-size research institution will use the services of an existing HTS facility rather than set up one for itself.

Methods for describing molecules in a manner more related to their biological activity would have the potential to enable modifications and research activities to progress in a more sufficient way. The goal for finding relevant, non-obvious, accurate bioisosteres is not lucking an interest. These methods broadly fall into two categories: knowledge-based approaches and computational techniques.

2. Application of the Graphical processing unit in the field of Bioisosterism

2.1 Parallel Computing

Even though supercomputers have kept a substantial lead over even the most sophisticated desktop machines in the continuous competition of hardware for speed, capacity, and robustness of computer platforms, the use of supercomputers has been considered a privilege to a limited number of people. In one important aspect — delivery of the technology to the fingertips of the largest number of peop In recent years, parallel computers, commonly known as "supercomputers", have demonstrated superior computational power, both in terms of speed and data volume capacity, in a variety of applications. Relying on the simple principle that doing several things at once is faster than doing them one after the other, Parallel computing platforms provide a solution to demanding and time-sensitive algorithms. Parallel computing is often invaluable when addressing problems that have an inherently complex structure, with examples varying from climate forecasting to n-dimensional modeling, Monte Carlo simulations, cryptography, etc [1]. As demonstrated further in this section, the linear algebra operations behind the quantum calculations for analytical evaluation of **bioisosterism** may employ an algorithm that is efficiently transferable to a parallel architecture, resulting in significantly improved execution timings.

Even though supercomputers have kept a substantial lead over even the most sophisticated desktop machines in the continuous competition of hardware for speed, capacity, and robustness of computer platforms, the use of supercomputers has been considered a privilege to a limited number of people. In one important aspect — delivery of the technology to the fingertips of the largest number of people — parallel computing has always been at a disadvantage. Large academic institutions, large corporations, and government organizations can afford to execute computing tasks on customized supercomputers. As the name entails, algorithms executed on parallel computers and implemented in MPI (Message Passing

Interface) have proven to be vastly superior in terms of time performance [2], which however comes at a high price: depending on the number of processors, and therefore computational potential, supercomputers can be priced anywhere upwards of several thousand dollars to build. Moreover, demand for computing frequently will be greater than the available processing time, requiring the process to be delayed in a scheduling queue. The resources and personnel required to establish and to keep a cluster operational are economically justifiable only in rare occasions.

To address the affordability problem, NVIDIA Corporation made available in the end of

2007 its proprietary platform for parallel programming, CUDA – Compute Unified Device Architecture [3]. The platform has two components: (1) hardware - the NVIDIA Graphics Processor Unit (GPU) on the graphics card; and (2) software - the programming interface to the GPU, provided by the CUDA language. As initially intended, the parallel computational capability would provide for efficient graphics rendering operations, where simple, independent algebraic calculations must be executed. The native carrier for graphics information is the **matrix**, stored in memory in the form of an *array*. Thus, graphics cards have gradually undergone a natural evolution to become robust platforms for matrix operations in parallel, a virtual requirement for all modern video-editing and gaming software.

An example of a problem that is suitable for a parallel implementation is the operation of matrix summation, A + B = C. The sum of elements in row *i* and column *j* of both matrices, A*ij* and B*ij* , is recorded in element C*ij* . This operation is *independent of the additions performed to compute the results of the remaining elements*, as demonstrated in Fig.1:

This operation is *independent of the additions performed to compute the results of the remaining elements*, (Fig. 1).

Theoretically, nothing prevents two operations from matrix addition from being executed at the same time, which is exactly what happens in practice when employing a parallel platform, where a distinct CPU is responsible for independently making each calculation and recording the result into memory [4].

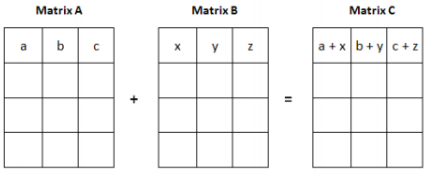

Fig. 1: Operations required for matrix addition

A similar approach, although slightly more complex – due to the interconnected information that is required to gather and store (memory reads and memory writes), in terms of the matrix elements, as seen in Fig. 2:

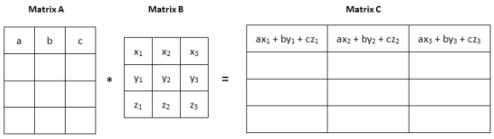

Fig. 2: Operations required for matrix multiplication

An element in row *i* column *j* in C is the sum of by-element products of row *i* from A and column *j* from B. Although in theory all elements in C can independently be calculated, for a large size matrix, even in multi-processor machines, there are not enough available processors to do all the necessary calculations in parallel. Instead, processors have to loop over the elements of the rows and columns of the matrices in order to produce the result of the multiplication. The following Fig. 3 is used to distinguish between the architecture of a CPU and a GPU:

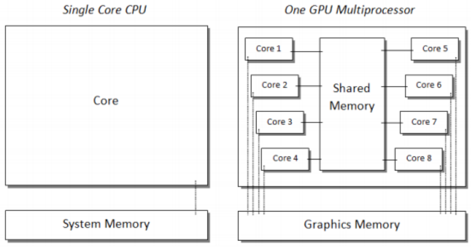

Fig. 3: Comparison between CPU / GPU

When data is prepared for manipulation by the GPU, it is divided into blocks, which are then independently assigned for execution on one of the multiprocessors. Each block can have as many as 512 threads running simultaneously. Unless data is structured properly into blocks that are of size a multiple of a *warp* (32), a number of the threads in a warp may be inactive during a cycle, which is a drawback when the goal is maximum speed of execution. [3] However, there is still a great advantage in parallel calculations, such as matrix-vector multiplication, demonstrated in Fig. 4:

The speed-up of the multiplication for the first step, when done in parallel, as compared to a serial computation, will be the lesser of the two – the number of threads in a vCPU, or the width of the matrix (i.e. for a matrix that is 50 elements wide, we can expect a speed-up of 50x).

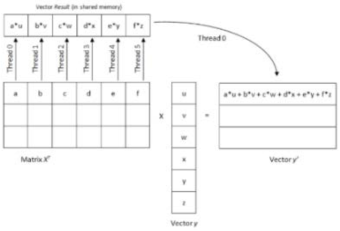

Fig. 4: Calculating matrix-vector multiplication in parallel

2.2. Uses of Linear Algebra in Chemistry

A multitude of works have shown the importance of matrix representation in chemistry. By examining just a few of them, one can note the importance of linear algebra for chemical applications, and therefore, apply the tools that pertain to linear algebraic calculations, to chemistry. The one tool, as explained in the previous section, is parallel algorithms for matrix operations, as executed on the Graphics Processing Unit.

2.2.1. Quantum (wave-function) representation of atoms and molecules

Ever since Schrödinger and De Broiglie first posed the principles of wave mechanics, and Heisenberg and Born – those of matrix mechanics – and subsequently, Schrödinger proved the two were equivalent (since the eigenfunctions of Hermitian operators are orthogonal, and can be normalized), it was made clear that linear algebra will play an important role in further research.

For example, wave-functions can be represented by vectors, and operators by matrices [6]. If we define the components of a state vector ψ:

$$\psi_i \equiv \langle u_i|\psi\rangle \qquad |\psi\rangle = \sum_i \psi_i|u_i\rangle$$

If we then define a matrix element:

$$O_{ij} \equiv \langle u_i|O|u_j\rangle$$

And use it as an operator acting on a wave-function:

$$|O\psi\rangle = O|\psi\rangle = O\sum_j \psi_j|u_j\rangle = \sum_j \psi_j O|u_j\rangle$$

Since an operator of a wave-function gives a wave-function, when the dot-product of the vector $\langle u_i|$ is included in the equation, we get:

$$(O\psi)_i = \langle u_i|O\psi\rangle = \sum_j \psi_j\langle u_i|O|u_j\rangle = \sum_j O_{ij}\psi_j$$

Which shows, that the formula for a state vector can simply be yielded through a multiplication of a matrix operator times another state vector [6]:

$$\begin{pmatrix} (O\psi)_1 \\ (O\psi)_2 \\ \cdots \\ (O\psi)_i \\ \cdots \end{pmatrix} = \begin{pmatrix} O_{11} & O_{12} & \cdots & O_{1j} & \cdots \\ O_{21} & O_{22} & \cdots & O_{2j} & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ O_{i1} & O_{i2} & \cdots & O_{ij} & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{pmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \cdots \\ \psi_j \\ \cdots \end{pmatrix}$$

In a similar manner, we can evaluate the product of two operators, O and P:

$$(OP)_{ij} = \langle u_i|OP|u_j\rangle = \sum_k \langle u_i|O|u_k\rangle\langle u_k|P|u_j\rangle = \sum_k O_{ik}P_{kj}$$

Which can be represented as a product of two matrices [6]:

$$\begin{pmatrix} (OP)_{11} & (OP)_{12} & \ldots & (OP)_{1j} & \ldots \\ (OP)_{21} & (OP)_{22} & \ldots & (OP)_{2j} & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ (OP)_{i1} & (OP)_{i2} & \ldots & (OP)_{ij} & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \end{pmatrix} =$$

$$\begin{pmatrix} O_{11} & O_{12} & \ldots & O_{1j} & \ldots \\ O_{21} & O_{22} & \ldots & O_{2j} & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ O_{i1} & O_{i2} & \ldots & O_{ij} & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \end{pmatrix} \begin{pmatrix} P_{11} & P_{12} & \ldots & P_{1j} & \ldots \\ P_{21} & P_{22} & \ldots & P_{2j} & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ P_{i1} & P_{i2} & \ldots & P_{ij} & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \end{pmatrix}$$

Thus, matrix operations are fundamental to expressing and evaluating electron energy states through their wave-function vectors.

2.2.2. Z-matrix as a descriptor of molecules

Another important aspect of molecular chemistry evaluation is the use of Z-matrices, which completely describe any atom in a molecule by its internal coordinates, by examining its atomic number, bond length, bond angle, and dihedral angle. One of the ways, in which parallel algorithms can alleviate calculations is the conversion of Z-matrix coordinates to Cartesian coordinates, as the problem is not trivial for macromolecules such as DNA, polymers, and proteins. Moreover, as one of the fundamental formats used for electronic storage and description of the molecules, being able to manipulate molecules derived from the Z-matrix form is invaluable to bioisosterism as well.

When given the molecule structure of Taxol, it becomes evident how complex any attempts in terms of bioisosterism manipulation can be without the use of computational tools:

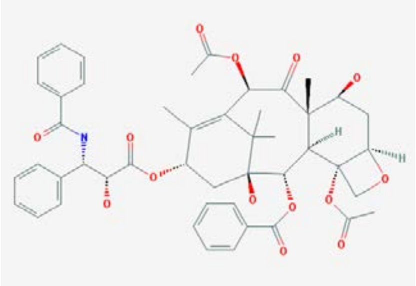

Fig. 5: Molecular structure of Paclitaxel / Taxol

The corresponding Z-Matrix form can only be comprehended through analytical modelling software, as it has 112 rows, corresponding to the number of atoms, in seven columns. Initially, such compounds could only be studied at a macro-chemistry level, with their reactivity construed only empirically. With the advance of universal file formats, such as the V2000, and algorithms, like the NERF (Natural Extension Reference Frame), which have parallelism potential, the application of the GPU can improve calculation times, and therefore, bioisosterism candidates, by order of magnitudes [7]. As Parsons et. al. suggest in the conclusion of their article, “(…)most of the NeRF method, even the filling out of the rotation matrix, is conveniently written in terms of vector operations like crossproducts and norms. This strongly suggests explicitly rewriting the algorithm to take advantage of vector processing opportunities, such as SIMD-type operations found on many modern math processors (…)” [7]. As the authors’ work was done in 2005, two years before the beta release of the CUDA programming language for nVidia GPUs, it is clear how their hypothesis will directly transfer to a SIMD platform environment, such as the CUDA nVidia has to offer.

2.2.3. Molecular symmetry analysis via matrix representations

For a specific class of molecules, which have symmetry properties (about an axis, a plane, a line or a point), linear algebra operations are an indispensable tool. When evaluating molecules and their transformations (symmetrical) in 3D-space, all calculations are done using matrices.

The following are three simple examples of how vector-matrix multiplication can be used to achieve symmetrical transformations in 2D [5]:

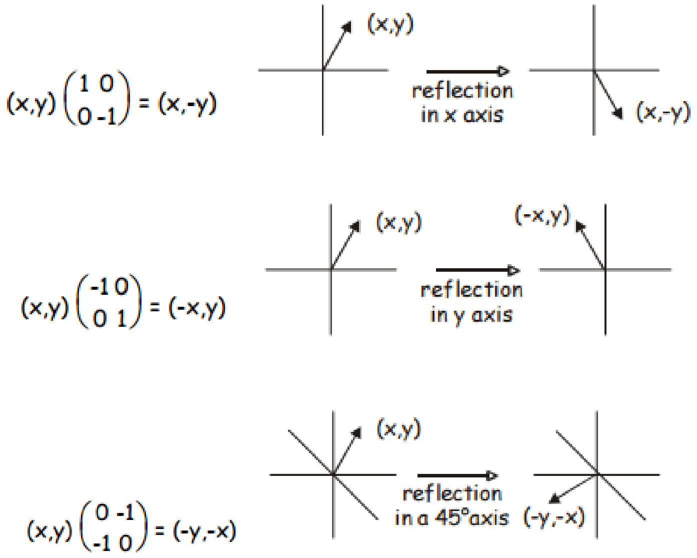

Fig. 6: Symmetrical transformations in 2D via matrix-vector multiplication

Similarly, we can achieve a custom rotation in 3D, around the x, y and z axes at an angle of Θ, by multiplying the coordinate vector by the following matrices [5]:

$$R_x(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \qquad R_y(\theta) = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \qquad R_z(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

For molecules that have symmetrical properties, operations requiring their transformations in space, in the search for bioisosterism, may be greatly sped-up by employing the GPU.

2.3. Bioisosterism by utilizing matrix functions and the GPU architecture

In conclusion, from the material presented in the previous sections, we can determine the importance of parallelism in bioisostere search algorithms – especially for molecules with complex structures. A rigorous analytical approach to examining bioisostere candidates by using any analytical descriptor (quantum wave-function electron field approach, symmetrical transformation approach, or other) could require trillions of linear algebra operations. The speed up of vector-matrix multiplication, matrix-matrix multiplication, and matrix inversion varies from 50-500x and above. To put this into perspective, any software that relies on the CPU for a specific bioisosteric algorithm (be it NERF or other), and takes one hour to deliver a full list of potential bioisosteres by utilizing the CPU, could deliver the same number of results in less than 10 seconds. When applying a "visual triage technique"- when the researcher needs to examine the bioisosteres visually, a difference between an 1-hour wait and a 10-second wait could prove to be of vast interest in making the process much more efficient.

Acknowledgments:

This work has been supported by the Program for career development of young scientists, Grant No 112/2016

Bibliography:

[1] Blaise Barney. An introduction to parallel computing. In https://computing.llnl.gov/tutorials/parallel comp/, USA, 2009. Lawrence Livermore National Library.

[2] Mark Esman. Kind of machine to run Stata. In http://www.stata.com/support/faqs/win/specs.html, USA, 2006. StataCorp, LP

[3] NVIDIA. Cuda programming guide. Pp. 1–98, 2701 San Tomas Expressway, USA, 2008. NVIDIA Corporation.

[4] Harry F. Jordan and Gita Alaghband. Parallel machines and computations. In Fundamentals of Parallel Programming, Marcia J. Horton, editor, pages 2–16, New Jersey, USA, 2003. University of Colorado, Prentice Hall

[5] Vinter A and Rose S, Molecular Field Technology and its Applications in Drug Discovery, Innovations in Pharmaceutical Technology, 23, pp. 14-18, 2007

[6] Bellenie BR, Barton NP, Emmons AJ, et al, Discovery and optimization of highly ligand-efficient oxytocin receptor antagonists using structure-based drug, Bioorganic and Medicinal Chemistry Letters, 19, pp. 990-994, 2009

[7] Dart MJ, Wasicak JT, Ryther KB, et al, Structural aspects of high affinity ligands for the a4b2 neuronal nicotinic receptor, Pharmaceutica Acta Hevetiae, 74, pp. 115, 2000

[8] T. Cheeseright, The Identification of Bioisosteres as Drug Development Candidates, Cresset BioMolecular Discove, 2007

[9] Comparative Molecular Field Analysis (CoMFA) Hugo Kubinyi BASF AG, D-67056 Ludwigshafen, Germany

Применение GPU в области биоизостеризма

Десислава Иванова

Институт информационных и коммуникационных технологий, 1113 София, Болгария

divanovaacomin@gmail.com

Резюме: Программные применения для 3D визуализации и оценки биоизостеричных молекул очень важны в биоизостеризме. Использование графических процессоров (GPU) для параллельной обработки операций на основе матриц - для расчетов квантовых векторов - приведет к многократному ускорению скорости этих вычислений. В этом вопросе GPU является фактором, который важен для улучшения методов оценки изследования и визуализации молекулярного сходства.

Ключевие слова: 3D визуализация, биоизостерические молекулы, програмные применения,
иновативные методы, биоизостеризм, GPU