

HPC em Simulações Moleculares

Paulo E. P. Burke, Tárcius Ramos

15 de junho de 2017

1 Dinâmica Molecular

A simulação da dinâmica de sistemas moleculares é uma importante ferramenta em diversos campos de pesquisa como física, química, engenharia de materiais e biologia molecular. Esta dinâmica envolve prever a posição espacial e energia de átomos contidos em um sistema ao longo do tempo. Esta dinâmica pode ocorrer em diferentes escalas de espaço e tempo, como exemplificado na Figura 1 para sistemas biológicos.

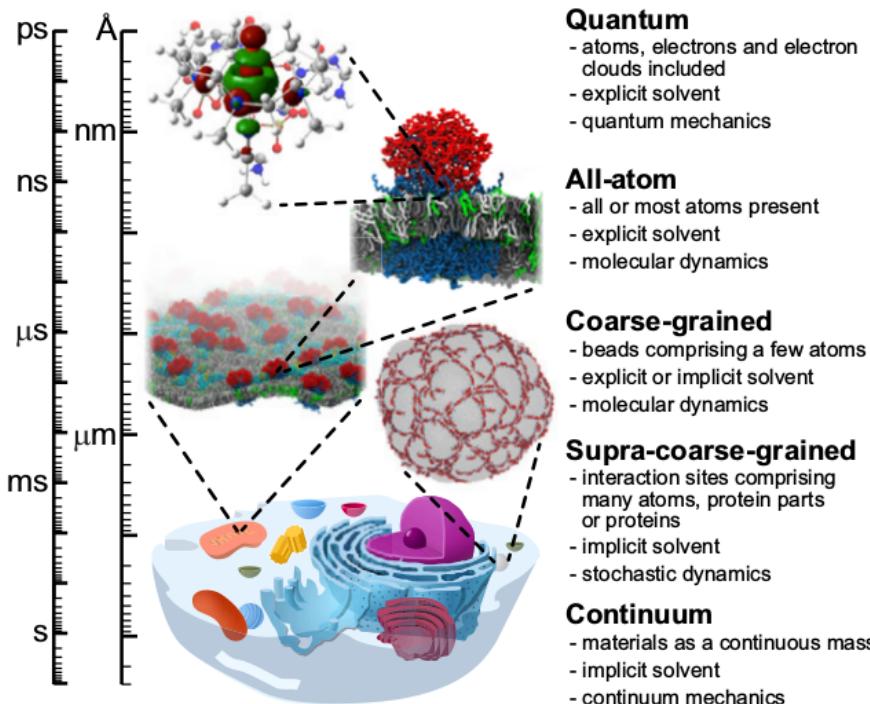


Figura 1: Escalas de espaço e tempo em simulações de dinâmica molecular [1].

Contudo, estas simulações são computacionalmente custosas, sendo este custo computacional uma barreira nas pesquisas que envolvem esta técnica para estudos de sistemas grandes. Para evitar tal impedimento em escala da simulação, os softwares desenvolvidos para tais aplicações vem empregando a computação paralela como uma solução para tal problema. Atualmente, praticamente todos os softwares de simulação de dinâmica molecular empregam computação paralela e distribuída em seus códigos, possibilitando o acompanhamento do aumento do poder computacional e portanto, possibilitando a simulação de sistema cada vez maiores.

2 Modelagem Molecular

As propriedades físico-química de uma substância indicam seus potenciais de aplicação tecnológica, que são tão diversos quanto o desenvolvimento científico. Essas propriedades representam o modo e a intensidade das interações intramoleculares e intermoleculares. As fases sólidas, líquidas e gasosas influenciam diretamente nas interações intermoleculares, ocasionando alterações em propriedades físico-químicas. Em fase gasosa, devido à baixa densidade, estas interações são fracas e podemos, frequentemente, considerar que a substância está isolada e livre no espaço. Este é o modelo do gás ideal. Enquanto em fase sólida esta interação intermolecular é intensa, caracterizada pela alta densidade, e aprisiona a substância numa determinada região. Nestas condições também é possível considerar um modelo para um sólido ideal. Na fase líquida, as moléculas da substância e do líquido (solvente) interagem fortemente e não estão fixas. Isto está relacionado com a densidade e o livre caminho médio, intermediários entre o sólido e o gás. Devido a essa complexidade, não existe um modelo de líquido ideal e o desenvolvimento de métodos capazes de incluir efeitos deste ambiente se fazem necessários. Do ponto de vista biológico, todos os sistemas de interesse estão majoritariamente em fase líquida e apresentam um grande número de átomos para serem simulados [2].

Uma forma de considerar as interações entre o soluto e o solvente seria através da propagação de equações de movimento considerando as interações como a contribuição de pares atômicos através de campos de força parametrizados, está técnica é conhecida como Modelagem Molecular [3]. Os campos de força buscam mimetizar o comportamento através de representações clássicas da carga, raio atômico, distância de ligação, etc.

Existem várias propostas de campos de força (AMBER, GROMOS, OPLS-AA, REAX, etc) e de métodos de integração (Verlet, Leap-Frog, Velocidade de Verlet, etc) das equações que regem o movimento dos átomos numa simulação. Todos os campos de força dependem de parâmetros que devem ser fornecidos no início da simulação. Independentemente do campo de força utilizado, o maior tempo de computação está no cálculo da energia total do sistema e das $3N$ componentes de força. Como cada um dos N átomos interagem com os $(N - 1)$ átomos restantes, isso resulta numa escalabilidade com N^2 . Geralmente confina-se o sistema de interesse numa caixa cúbica que é replicada em todas as direções.

Para melhorar a performance computacional algumas aproximações podem ser adotadas. Ao invés de todos os átomos interagirem entre si, pode-se definir que cada átomo irá interagir somente com os átomos dentro de uma esfera. O maior raio possível é metade do lado do cubo, nessa condição reduzimos a quantidade de interações pela metade. Ainda pensando no aumento de performance, nessas regiões se encontram a possibilidade de paralelização. Cria-se listas chamadas de “primeiros vizinhos” (que devem ser atualizadas periodicamente) contendo informações de quais átomos interagem com outros. Uma vez que estas listas estão definidas, essas interações são facilmente paralelizadas com OpenMP, MPI ou em dispositivos aceleradores.

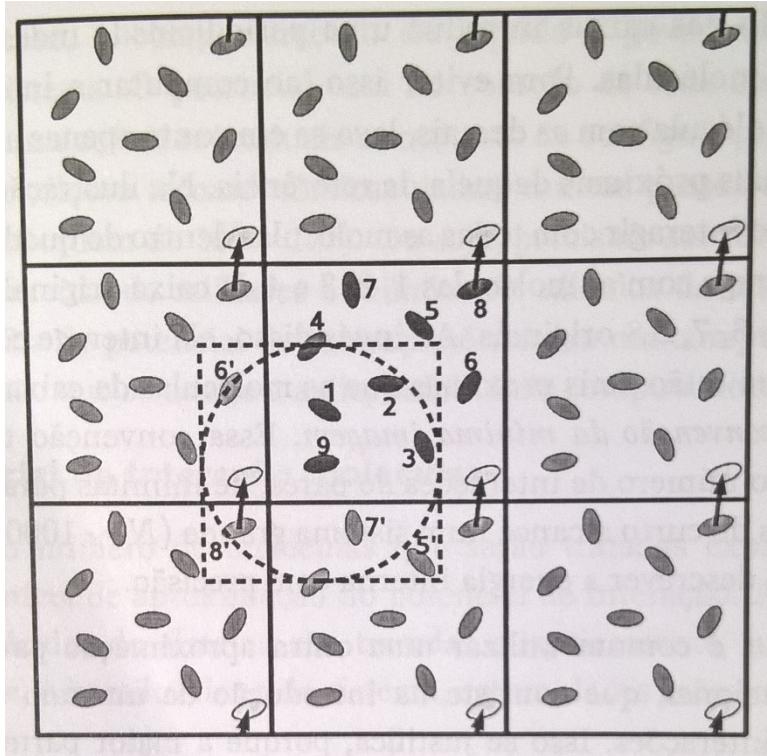


Figura 2: Replicação das caixas cúbicas e condição de contorno periódica. O quadrado tracejado representa a convenção da mínima imagem e a circunferência o raio de corte máximo.

3 Capacidade de Simulação

A simulação de dinâmicas moleculares, como citado na seção anterior, é um problema que cresce de forma quadrática com o número de átomos. Dessa forma, o tamanho do sistema a ser simulado é limitado pela capacidade computacional

disponível. A Figura 3 mostra exemplos de sistemas biológicos que eram capazes de serem simulados com o poder computacional disponível ao longo dos anos, desde a década de 90 até o ano de 2011 [4].

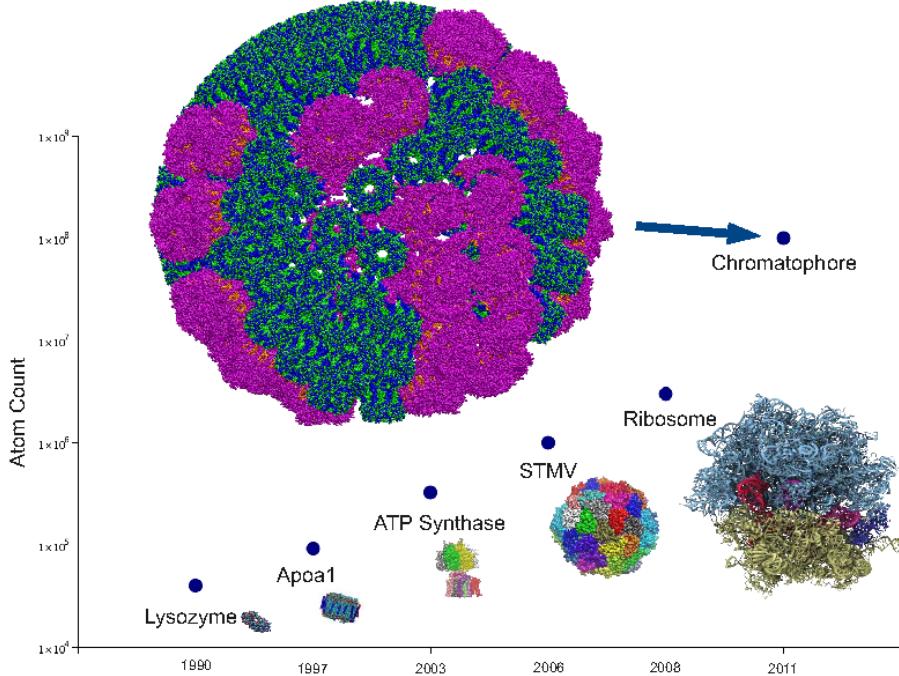


Figura 3: Capacidade de simulação em número de átomos ao longo do tempo.

É possível observar que a curva de quantidade de átomos possíveis de serem simulados ao longo do tempo se assemelha à curva da Lei de Moore. Em 2011, a capacidade de simulação já se aproxima de 10^9 átomos, permitindo-se estudar sistemas biológicos muito mais complexos como cromatopóros.

4 Desempenho dos Atuais Softwares

Existem muitos softwares disponíveis para realizar simulações moleculares. Os programas mais utilizados em simulações moleculares são: o Gromacs, o NAMD e o LAMMPS. Abaixo apresentaremos alguns *benchmarks* realizados com estes programas enfatizando a paralelização. A comparação entre os diferentes programas não será realizada pois os testes utilizados pelos desenvolvedores são diferentes e executam em clusters com diferentes comparações. Assim, somente o ganho de performance nas diferentes plataformas de paralelização serão comparadas para cada programa individualmente.

4.1 Gromacs

Desenvolvido inicialmente em Fortran 77 teve seu código reescrito em C em 1991 e atualmente suporta paralelização com OpenMP, MPI e CUDA¹. Dados fornecidos pelo desenvolvedor em junho de 2015 para a versão 5.0 do programa. Os testes foram realizados em 4 clusters e são medidos em nanosegundos simulados por dia²:

- Vesta: 6 cores por nó - PowerPC A2
- Beskow: 16 cores por nó - Xeon E5-2698v3
- Piz Daint: 8 cores por nó - Intel Xeon E5-2670 + 1 NVIDIA Tesla K20X GPU
- Hydra: 10 cores por nó - Intel Xeon E5-2680 + 2 NVIDIA K20X GPUs

Observa-se que a arquitetura e a performance do cluster são essenciais para a obtenção de melhores resultados. Comparando Beskow e Vesta, dois clusters formados por CPUs vemos que o aumento Beskow aumenta em torno de 100 ns/dia saindo de 512 para 2048 processadores enquanto o Vesta incrementa 25 ns/dia. Quando comparamos a escalabilidade com GPUs vemos um comportamento muito similar entre os dois clusters com destaque de melhor performance para o de menor capacidade. Isso indica que este programa não escala bem para múltiplas GPUs.

Como foi comentado na seção de Modelagem Molecular, a paralelização é realizada no cálculo da energia e ocorre após a definição da lista de primeiros vizinhos (veja o esquema na Figura 5 [5]).

4.2 NAMD

Desenvolvido diretamente em C++ no ano de 1995, recebeu a implementação em GPU em 2011 e em 2014 passou a suportar a aceleradora Xeon PHI da Intel³.

Temos dois testes⁴ realizados na Figura 6, um com 21×10^6 e outro com 224×10^6 átomos. Observa-se que para o sistema com menor quantidade de átomos a performance obtida fica distante do teórico esperado e que a inclusão da aceleradora Xeon PHI não gerou ganho de performance. Para os dois sistemas a melhor performance foi obtida com GPUs.

4.3 LAMMPS

Inicialmente desenvolvido em Fortran + MPI em 1999 foi reescrito em C++ em 2004, teve a implementação em GPU no ano de 2010 e em Xeon PHI em 2015⁵.

¹<http://www.gromacs.org>

²<http://www.gromacs.org/@api/deki/files/240/=gromacs-5.0-benchmarks.pdf>

³<http://www.ks.uiuc.edu/Research/namd/>

⁴<http://www.ks.uiuc.edu/Research/namd/performance.html>

⁵<http://lammps.sandia.gov/>

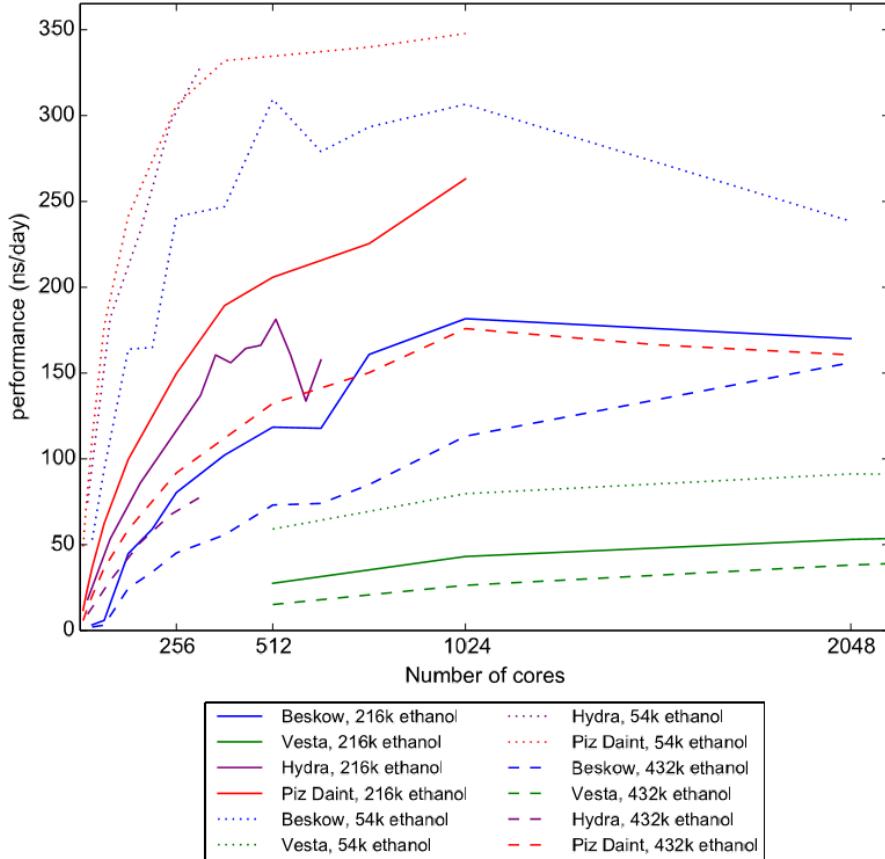


Figura 4: Benchmark GROMACS

Selecionamos este teste⁶ na Figura 7 para demonstrar a capacidade do programa na paralelização em um único servidor com configurações acessíveis sem a necessidade de um supercomputador. Variou-se a quantidade de processadores e de aceleradores utilizando duas implementações para GPUs. Observa-se que em 64 mil átomos a performance estabilizou, exceto para o teste com duas placas gráficas que mesmo com 2048 mil átomos apresentava um crescimento significativo de performance.

Diferentemente do programa NAMD o LAMMPS é altamente escalável para múltiplas GPUs. Também é interessante destacar que os desenvolvedores incentivam a utilização do *Hyper-Threadin* (HT) na execução da simulação, eles observaram um ganho entre 10% e 30% com a troca de contexto no HT⁷.

⁶<http://lammps.sandia.gov/bench.html>

⁷http://lammps.sandia.gov/doc/accelerate_intel.html

5 Super Computadores Disponíveis

Quanto maior o sistema de estudo, maiores são as necessidades computacionais. Dentro deste aspecto, listamos as principais configurações de alguns supercomputadores disponíveis para fins acadêmicos. Lembramos ainda que geralmente todo grupo de pesquisa dispõem de alguns recursos computacionais em suas unidades, mas raramente eles estão montados numa infraestrutura que otimize a utilização em diferentes nós.

Na Universidade Estadual Paulista (Unesp) está instalado o cluster GRID UNESP. A Universidade de São Paulo (USP) implementou dois clusters o Águia formado somente por CPUs e o Lince que contem GPUs da Nvidia como aceleradores.

Existe ainda o Sistema Nacional de Processamento de Alto Desempenho (SINAPAD) que fornece recursos para a instalação de Centros Nacionais de Processamento de Alto Desempenho (CENAPADs) em universidades em todo o Brasil. O maior projeto do SINAPAD é o cluster Santos Dumont instalado em Petrópolis no Rio de Janeiro que conta com servidores dedicados a CPUs, a aceleradores (tanto GPUs Nvidia quanto Intel Xeon Phi) e um servidor com 6TB de memória RAM para atender a demanda de programas de memória compartilhada.

Para a utilização de suas capacidades é necessário a submissão e aprovação de um projeto de pesquisa vinculado a uma instituição de ensino superior ou sob a supervisão de algum centro de pesquisa. Abaixo é possível encontrar informações mais detalhadas destes supercomputadores.

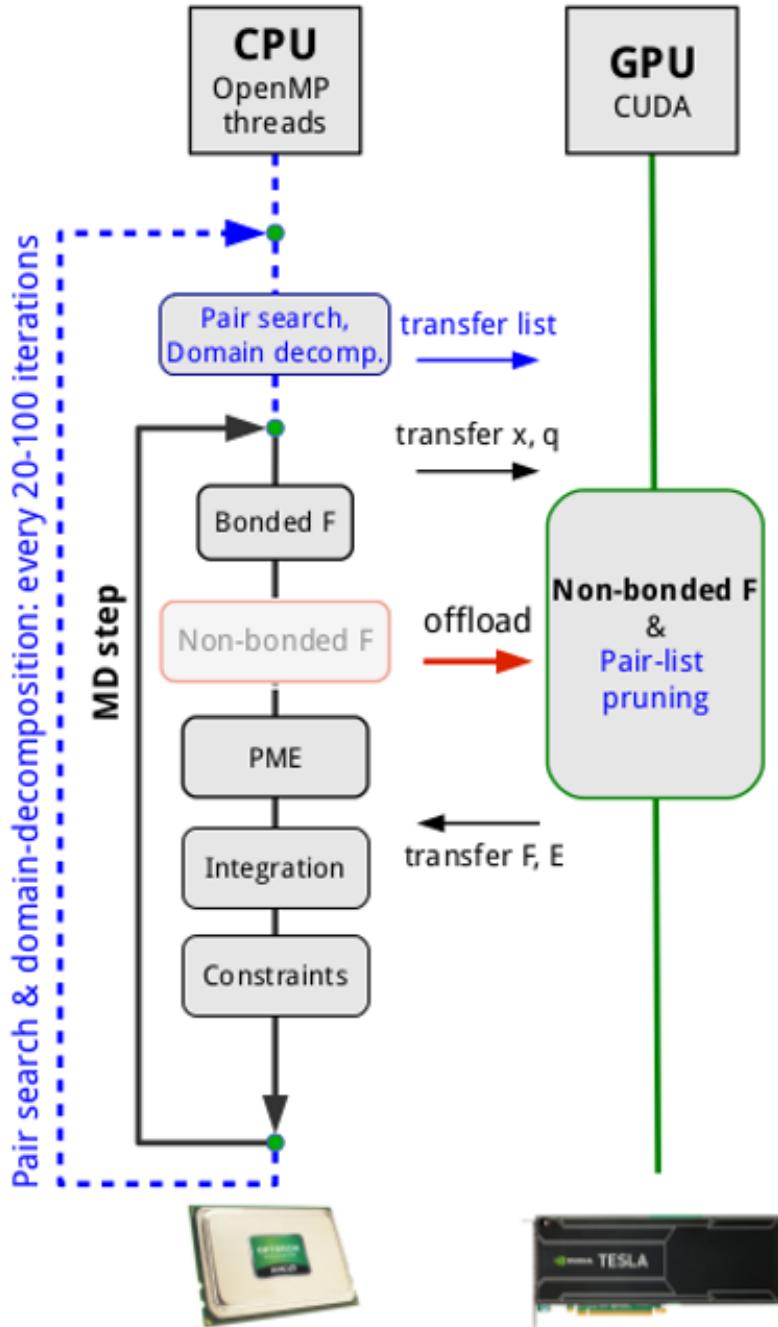


Figura 5:

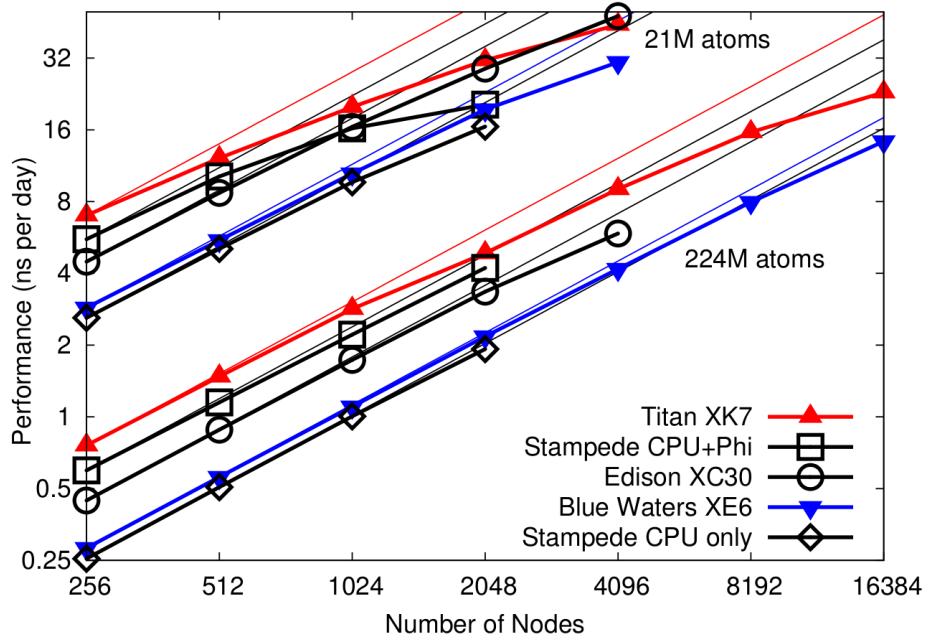


Figura 6:

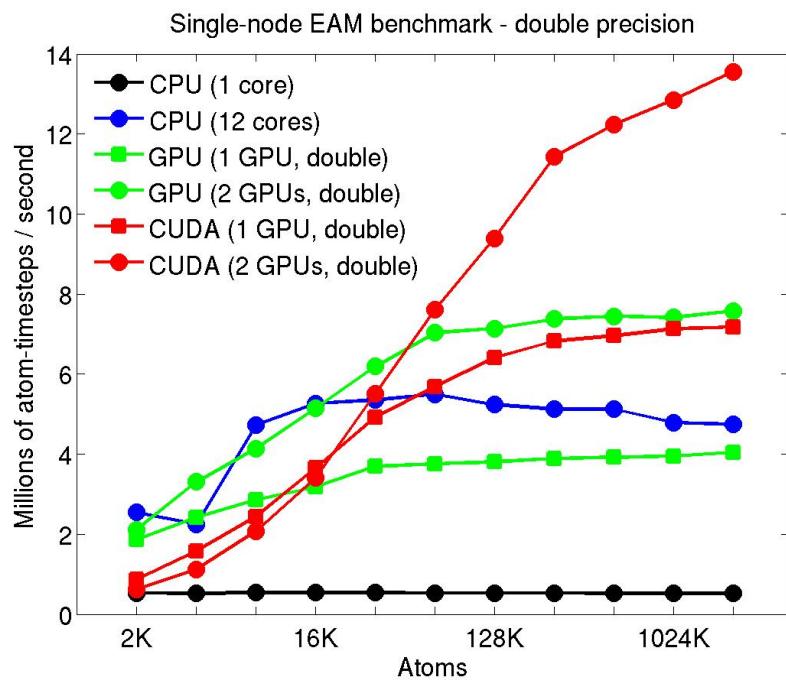


Figura 7:



Figura 8: GRID UNESP



Figura 9: Santos Dumont – LNCC

- GRID UNESP

256 servidores SUN X4150
2048 cores (Intel Xeon 2.83 GHZ)
4096 GB de memória RAM (2 GB por core)
Inteconexão Infiniband 4X DDR (20 Gbps)

- USP CLOUD

AGUIA

64 servidores
1280 cores (Intel Xeon 2.40 GHZ)
32768 GB de memória RAM (25 GB por core)

LINCE

32 servidores
512 cores (Intel Xeon 2.40 GHZ)
4128 GB de memória RAM (8 GB por core)
2 placas Tesla K20m
Blue Gene / Q
1024 servidores
16384 cores
16384 de memória RAM (16 GB por core)

- SINAPAD

Santos Dumont – LNCC

756 servidores
18144 cores (Intel Xeon 2.40 GHZ)
48384 GB de memória RAM (2.7 GB por core)
198 servidores com 2 placas Nvidia K40
54 servidores com 2 placas Xeon PHI 7120
1 servidor com 240 cores e 6 TB de RAM.

CENAPAD - Centro Nacional de Processamento de Alto Desempenho

9 Unidades em todo o Brasil.

6 Expectativas para *Exascale*

As aplicações de dinâmica molecular aliada à capacidade computacional atual já permitem o estudo de grandes sistemas moleculares em diversas áreas e especialmente na biologia molecular. O advento do *Exascale* terá implicações muito grandes nessa área de pesquisa. A possibilidade de simular grandes sistemas abre portas para entender, por exemplo, interações entre quaisquer duas proteínas que se conheça a estrutura tridimensional. Isto terá um grande impacto em pesquisas biológicas de diversas áreas, podendo substituir experimentos custosos tanto financeiramente quanto em mão de obra. O entendimento e o desenvolvimento de curas para doenças será acelerado de forma à contribuir direta e indiretamente para toda a sociedade [6].

7 HACC

Como um contraponto às aplicações microscópicas de aplicações de modelagem computacional apresentadas, destaca-se o Hardware/Hybrid Accelerated Cosmology Code (HACC)⁸. Com este código realizou-se uma simulação cosmológica com condição inicial em 50 milhões de anos após o Big Bang até os dias atuais (13.8 bilhões de anos). Para realizar essa tarefa muitas análises e melhorias foram realizadas para se obter entre 90% e 99% de eficiência de paralelização. Nesta etapa, o código foi otimizado especificamente para cada hardware observando-se limites de vetorização, “indireção”, estruturas de dados complexas, falta de encadeamento e listas de interações curtas. Por ser uma estrutura para diversos hardwares foi necessário implementar uma estrutura multi-algoritmica com MPI para a interface entre os servidores e CUDA, OpenCL e OpenMP para a estrutura nodal.

O desenvolvimento deste código mostra que com dedicação e conhecimento sobre a estrutura do hardware é possível obter uma eficiência de paralelização muito próxima de 100%.

Referências

- [1] H. I. Ingolfsson, C. Arnarez, X. Periole, and S. J. Marrink, “Computational microscopy of cellular membranes,” *Journal of Cell Science*, vol. 129, no. 2, pp. 257–268, 2016.
- [2] M. P. Allen and D. J. Tildesley, “Computer Simulation of Liquids,” *Liquids, Oxford University Press, New York*, vol. 18, no. 195, p. 385, 1987.
- [3] N. Morgan and K. Coutinho, *Métodos de Química Teórica e Modelagem Molecular*. 2007.

⁸<https://cacm.acm.org/magazines/2017/1/211098-hacc/fulltext>

- [4] C. Mei, Y. Sun, G. Zheng, E. J. Bohm, L. V. Kale, J. C. Phillips, and C. Harrison, “Enabling and scaling biomolecular simulations of 100 million atoms on petascale machines with a multicore-optimized message-driven runtime,” *2011 International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*, pp. 1–11, 2011.
- [5] S. Páll, M. J. Abraham, C. Kutzner, B. Hess, and E. Lindahl, “Tackling exascale software challenges in molecular dynamics simulations with GROMACS,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8759, pp. 3–27, 2015.
- [6] L. Wolf and G. W. Pieper, “Biology and Medical Research at the Exascale,” *SciDAC Review*, vol. 16, pp. 32–39, 2010.