

Um Modelo Computacional para Simulação de Interação de Spins em Elementos e Compostos Magnéticos

João Paulo Peçanha Navarro de Oliveira- joapaulo@ice.ufjf.br

Alessandra Matos Campos- amcampos@ice.ufjf.br

Patrícia Cordeiro Pereira Pampanelli- patricia.pampanelli@ice.ufjf.br

Marcelo Lobosco- marcelo.lobosco@ice.ufjf.br

Marcelo Bernardes Vieira- marcelo.bernardes@ufjf.edu.br

Departamento de Ciência da Computação

Sócrates de Oliveira Dantas- dantas@fisica.ufjf.br

Departamento de Física

Universidade Federal de Juiz de Fora, Cidade Universitária, Terceira Plataforma,

CEP: 36036-330 - Juiz de Fora, MG, Brasil

Resumo. *O estudo de fenômenos magnéticos em escala nanométrica é imprescindível para o desenvolvimento de novas tecnologias e materiais. Este estudo pode ser facilitado com o emprego de simuladores, que oferecem ao pesquisador a possibilidade de criar e alterar as condições de estudo de modo simples e rápido. Neste trabalho apresentamos um modelo físico-matemático-computacional para a simulação de spins em estruturas tridimensionais genéricas formadas por átomos com propriedades magnéticas. Este modelo é em seguida implementado através do simulador batizado de Monte Carlo Spin Engine. O simulador permite ao usuário definir condições de simulação, bem como a geometria do objeto a ser simulado. Detalhes visuais e numéricos gerados a partir da modelagem computacional proporcionam ao cientista uma importante ferramenta que auxilia no melhor entendimento de processos físicos.*

Abstract. *The study of magnetic phenomena in nanometer scale is essential for the development of new technologies and materials. This study can be facilitated with the use of simulators which offer to the researcher the ability to quickly and easily create and modify the conditions of the experiments. This work presents a physical-mathematical-computational model for the simulation of spins in tridimensional structures formed by atoms with magnetic properties. The Monte Carlo Engine Spin simulator implements this model. The simulator allows the user to define simulation conditions as well as the geometry of the object being simulated. Visual and numerical data generated by the simulator are important tools to better understand the physical processes.*

Keywords: simulação computacional, interação de spins, simulação de materiais ferromagnéticos.

1. Introdução

Os fenômenos magnéticos são amplamente utilizados no desenvolvimento de novas tecnologias, como sistemas de geração e distribuição de energia, dispositivos eletrônicos e de telecomunicações, e sistemas de conversão eletromagnética, dentre muitas outras. Dispositivos que utilizam esses fenômenos estão muitas vezes presentes em nossas atividades cotidianas,

muito embora sem que muitos se dêem conta disso, como é o caso dos discos rígidos dos computadores, cartões de crédito, televisores, aparelhos de videocassete e dos exames de ressonância magnética.

Para uma melhor análise e compreensão do comportamento dos fenômenos magnéticos, é imprescindível o estudo destes em escala nanométrica. Foi assim que os Físicos Albert Fert e Peter Grünberg descobriram, em trabalhos simultâneos e independentes, o efeito de Magnetoresistência Gigante, trabalho que lhes valeu o prêmio Nobel de Física em 2007. A descoberta deste fenômeno proporcionou um aumento da ordem de 100 vezes na capacidade de armazenamento dos discos rígidos. Além do mais, a própria origem do magnetismo está associada a duas propriedades dos elétrons em escala nanométrica: a) o momento angular, associada ao movimento destes ao redor do núcleo atômico; e b) *spins*, associada a forma como os elétrons ocupam os níveis de energia no átomo.

O estudo de fenômenos magnéticos em escala nanométrica pode ser facilitada com o emprego de simuladores. Estes são capazes de fornecer um conjunto detalhado de informações do comportamento dos fenômenos magnéticos quando sujeitos às mais diversas situações, que podem ser criadas e alteradas livremente pelo cientista. Detalhes visuais e numéricos gerados a partir da modelagem computacional proporcionam ao cientista uma importante ferramenta que auxilia no melhor entendimento de processos físicos.

Neste trabalho apresentamos um modelo físico, matemático e computacional para simulação da interação de *spins* em estruturas tridimensionais genéricas formadas por átomos com propriedades magnéticas. O objetivo do simulador é fornecer instrumentos para que Físicos analisem o comportamento dos *spins* de um objeto sob a ação de um campo magnético externo uniforme. As diretrizes que guiaram a construção do simulador foram a eficiência no cálculo das interações dos *spins* e a facilidade de análise do objeto simulado através de computação gráfica.

Nosso trabalho está organizado como se segue. Na Seção 2.0 apresentamos o modelo físico-matemático que guiará a construção de nosso simulador. Na Seção 3.0 apresentamos o modelo computacional utilizado na implementação do simulador, batizado de *Monte Carlo Spin Engine*, além de alguns resultados experimentais obtidos a partir de simulações nele executadas. A Seção 5.0 apresenta os trabalhos correlatos, enquanto a Seção 6.0 apresenta propostas de trabalhos futuros.

2. Modelo Físico-Matemático

Podemos classificar os materiais de acordo com a sua reação diante da influência de um campo magnético externo, aplicado de acordo com sua susceptibilidade $\chi = M/H$ (H é o campo externo e M a magnetização), criando com isso várias fases: ferromagnético e ferromagnética ($\chi \gg 1$); diamagnética ($\chi < 1$); paramagnética ($\chi > 0$); antiferromagnética (χ pequeno) e materiais supercondutores (dos Santos Cabral Neto, 2004). Em especial, neste trabalho focamos na modelagem dos elementos com propriedades ferromagnéticas, onde a interação entre seus átomos faz com que todos os momentos magnéticos dos mesmos tendam a dispor-se na mesma direção e sentido. Uma importante propriedade desses elementos é o fato de seu campo magnético local ser causado pelos seus *spins*.

Em mecânica quântica, o *spin* (\mathbf{S} – vetor 3D) de um átomo refere-se às possíveis orientações que partículas subatômicas (prótons, elétrons, nêutrons e alguns núcleos atômicos) têm quando estão sob ação, ou não, de um campo magnético externo. O *spin* não possui uma interpretação clássica, ou seja, é um fenômeno estritamente quântico. Podemos representá-los matematicamente com vetores orientados, nos permitindo, com isso, incrementá-los em formulações que nos retornam algumas características do sistema estudado.

Considere um sistema fechado composto de uma estrutura molecular ou cristalina única,

que pode tomar qualquer forma geométrica básica conhecida, como esferas, cilindros e cubos. Os átomos desta estrutura são modelados como pontos no espaço providos de uma carga não fixa \mathbf{S}_i que representa seu *spin*. Estes são igualmente espaçados em uma grade regular de tamanho variável, tendo como única influência externa um campo magnético (\mathbf{H}) inerentemente vetorial.

Estruturas magnéticas como estas, quando sob a influência de um campo magnético externo (\mathbf{H}) têm a tendência de orientar seus *spins* de acordo com a direção do mesmo, acompanhada de uma natural queda de temperatura. Ao longo do tempo esses *spins* vão girando, buscando preferencialmente o ajuste de direção com o campo. Essa mudança nos garante que a energia total do sistema varie a cada giro desta propriedade. Uma vez que cada átomo possui uma energia própria e posição espacial bem definida, é possível calcular a energia total do sistema ao longo do tempo.

Para N dipolos localizados em uma grade de simulação regular e igualmente espaçados, a energia total E_t das interações é dada por:

$$E_t = \frac{A}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^N \left\{ \frac{\mathbf{S}_i \cdot \mathbf{S}_j}{|\mathbf{r}_{ij}|^3} - 3 \frac{[\mathbf{S}_i \cdot \mathbf{r}_{ij}][\mathbf{S}_j \cdot \mathbf{r}_{ij}]}{|\mathbf{r}_{ij}|^5} \right\} - J \sum_{\substack{i,k=1 \\ i \neq k}}^N \mathbf{S}_i \cdot \mathbf{S}_k - \sum_{i=1}^N D(\mathbf{S}_i \cdot \mathbf{H}) \quad (1)$$

onde \mathbf{S}_i denota o *spin* da partícula analisada, $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ é o vetor posição que separa os átomos i e j , A corresponde à intensidade de interação dipolo-dipolo, J ao fator ferromagnético do elemento em questão, \mathbf{H} ao campo externo uniforme e D ao seu fator. O primeiro termo da expressão é chamado de interação dipolo-dipolo. Este termo de longo alcance é caracterizado computacionalmente por um alto custo de execução, pois a energia de cada átomo é calculada de um para todos, ou seja, para que a energia individual de um átomo, em especial, seja computada, é necessário o acesso à posição de todos os outros, sem exceção. Estabelecemos a este termo, então, complexidade computacional de $\mathcal{O}(n^3)$, se tornando com isso a fatia de computação mais cara de toda a expressão.

O segundo termo refere-se a interação ferromagnético entre os átomos. Por se tratar de uma interação de curto alcance, os elementos do sistema que exercem influência direta no resultado dessa parcela serão apenas os k átomos mais próximos ($k=1,2,3,\dots,6$) para uma rede quadrada regular. Já o terceiro termo da expressão reporta à influência que o campo externo exerce sobre cada átomo, tendo \mathbf{H} como o vetor que representa a sua direção.

Com todos os dados devidamente configurados, temos como resposta um E_t que corresponde à energia total do sistema. Uma análise minuciosa desta energia ao longo da simulação e em função da temperatura à qual o sistema esteja sujeito pode nos fornecer um dado muito importante: onde ocorre a transição de fase de um comportamento magnético para não-magnético. O comportamento não-magnético é associado ao momento em que o sistema sofre um decaimento abrupto no somatório total da energia. Assim, o principal objetivo da resolução numérica deste cálculo é encontrar, para uma determinada configuração, o E_t que corresponde ao valor crítico do sistema.

A mudança de fase de um sistema está diretamente ligada ao estado de energização em que ele se encontra. Contudo, existem outros fatores que influenciam a faixa de temperatura em que ocorre a transição entre o comportamento magnético e o não-magnético como, por exemplo, a magnetização interna do objeto. A variação na magnetização depende não apenas do material escolhido, mas também da configuração (por exemplo, térmica e magnética) a ele imposto. Adicionalmente, no caso de elementos com propriedades ferromagnéticas, temos que seus *spins* também exercem influência direta nesta propriedade; neste caso, para N átomos, temos:

$$M_t = \sum_{i=1}^N S_i \quad (2)$$

Onde M_t corresponde a magnetização total, S_i indica o *spin* do átomo i .

3. Modelo Computacional

Nesta Seção, será descrito como a aplicação foi modularizada, o algoritmo implementado e o processo de geração visual dos dados. A aplicação foi chamada de *Monte Carlo Spins Engine*, ou MCSE. A MCSE, desenvolvida na linguagem C, segue o modelo de programação sequencial e é dividida em duas partes distintas. Uma é responsável pela visualização dos resultados e a outra é responsável pela execução do algoritmo de Monte Carlo no qual a simulação é baseada. A interface que une as duas partes é a estrutura de dados chamada *matrix*. Esta estrutura foi criada com o propósito de mapear o arranjo de *spins* em uma grade tridimensional e armazenar as informações correspondentes a cada um de seus átomos na posição correspondente da grade.

3.1 Método de Monte Carlo

O método de Monte Carlo (MMC) obtêm aproximações numéricas para funções complexas através da conversão de um modelo físico ou matemático em um modelo estatístico. Os resultados são gerados sobre uma distribuição de probabilidades e a amostra obtida é usada para aproximar a função de interesse.

O MMC foi desenvolvido por Nicholas Metropolis e Stanislaw Ulam durante a segunda guerra mundial para estudar a difusão de nêutrons durante o fenômeno de fissão nuclear. Em 1953, Nicholas Metropolis desenvolveu o algoritmo que leva seu nome com base no MMC a fim de estudar o comportamento estatístico dos gases. Desde então este algoritmo é muito utilizado pela Física.

De maneira geral, o algoritmo de Metropolis determina valores esperados de propriedades de um sistema em simulação calculando-se uma média sobre uma amostra, que é obtida através da geração de números aleatórios. O algoritmo é concebido de modo a se obter uma amostra que siga a distribuição de Boltzmann (Metropolis et al., 1953): assim, o sistema a ser simulado deve se encontrar em temperaturas diferentes de zero e o valor da energia para cada partícula do sistema deve ser conhecido.

3.2 Implementação do algoritmo de Metropolis

O algoritmo de Metropolis (Fig. 1) implementado na MCSE é composto pelas seguintes etapas:

- Definição das condições físicas iniciais do sistema (arranjo espacial dos *spins*);
- Iniciação da grade;
- Escolha arbitrária de um *spin* a mudar de direção
- Mudança arbitrária na direção de um *spin*;
- Cálculo da nova energia total do sistema;
- Validação do resultado.

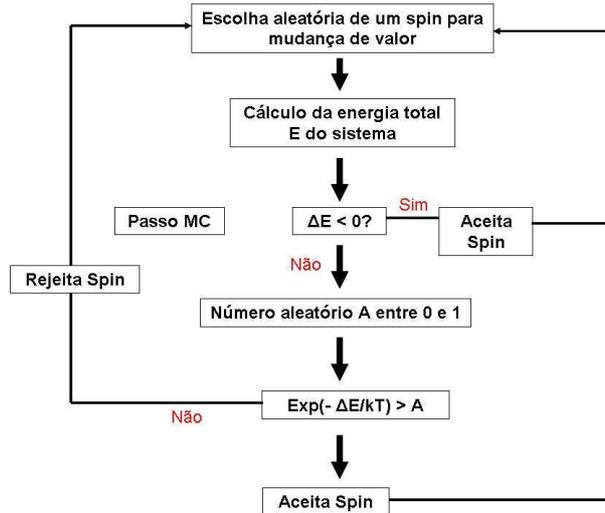


Figura 1: Fluxograma do algoritmo de Metropolis onde ΔE é a variação de energia obtida (antes e após a mudança do *spin*).

No passo 1 o usuário configura os parâmetros de entrada da aplicação através da interface para simulação. Nela, os valores da temperatura do sistema (Kt), fator da energia dipolo-dipolo (A), fator ferromagnético (J), o campo magnético externo (H) e o seu fator D , e a frequência com que os quadros são atualizados. O campo magnético externo é modificado por um controle que permite rotacioná-lo no espaço que compreende a grade inicial exibida na janela de visualização dos resultados. As dimensões da grade e o valor inicial dos *spins* são manipulados diretamente no código fonte.

Uma vez configurado o sistema, é passada à etapa 2 que consiste em iniciar a grade de simulação. Isso se dá através de quatro ações. Primeiro será feita a conversão do índice triplo de acesso à grade em um índice único, ou seja, a matriz, que é tridimensional, será acessada de maneira unidimensional. Feito isso, é verificado se a partícula visitada fará parte do arranjo de *spins*, isto é, se ela faz parte do suporte do objeto a ser analisado. A geometria do objeto é definida de forma implícita (Velho et al., 2002). A penúltima ação é atribuir o valor zero a energia inicial da partícula e, por último, é verificado se há algum valor inicial de *spin* a ser atribuído a todos os elementos. Se não há *spin* a ser atribuído, os valores de *spin* serão gerados aleatoriamente.

Na etapa 3, inicia-se a escolha aleatória de uma partícula cujo valor de *spin* será modificado. Este processo deve garantir que a probabilidade de sortear cada partícula seja uniforme. Para que esta condição ocorra utilizamos o algoritmo *Mersenne Twister* cuja periodicidade é de 2^{19937} (Matsumoto & Nishimura, 1998). A partícula escolhida recebe um novo *spin* também aleatoriamente distribuído no espaço de vetores unitários S^2 .

Para cada troca de *spin* é necessário calcular a energia total do arranjo de *spins* (Eq. 1). Esta é a etapa de maior custo computacional. Na validação da troca de *spin*, se a nova energia total do sistema for menor do que a energia total anterior, a condição de aceitação estará satisfeita e a troca de *spin* é implementada imediatamente. Porém, se a nova energia total é maior do que a anterior, possivelmente a temperatura do arranjo de *spins* estará influenciando o cálculo e essa troca de *spin* somente será aceita se satisfizer a seguinte condição:

$$e^{(-\Delta E/Kt)} > A \quad (3)$$

sendo que A é um valor aleatório no intervalo $[0, 1]$. Se este resultado for maior do que o número gerado então a troca de *spin* é implementada. Caso contrário, a troca é rejeitada e o *spin* anterior é reatribuído à partícula sorteada.

3.3 Monte Carlo Spin Engine

A MCSE gera gráficos para permitir a análise da variação de energia e da magnetização do arranjo de *spins* durante a simulação. Essas informações são de fundamental importância para uma análise realista e precisa do comportamento do sistema, dadas condições como temperatura, forma geométrica, campo externo no qual o sistema está imerso, dentre outras.

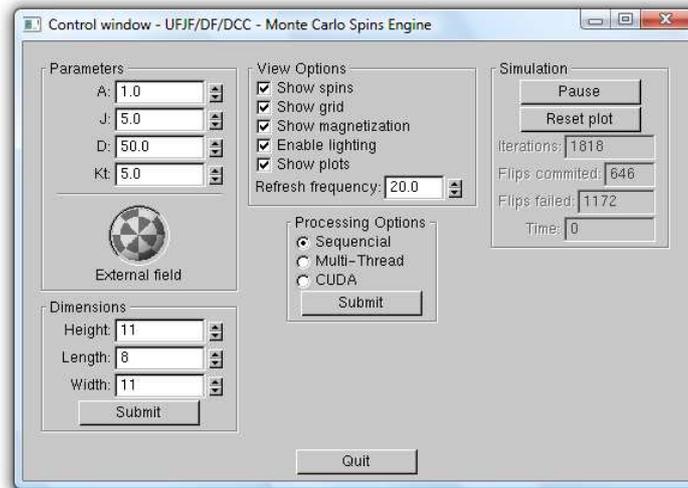


Figura 2: Em destaque a interface da MCSE

Na natureza podemos encontrar diversos elementos com propriedades ferromagnéticas. Para que possamos simular com precisão o comportamento de um elemento com propriedades ferromagnéticas em nosso simulador, faz-se necessário ajustar cada um dos parâmetros de simulação citados na Equação 1. O Ferro, Cobalto e Níquel são exemplos de elementos ferromagnéticos, e muito embora sejam de uma mesma família, apresentam algumas singularidades que influenciam no resultado final da simulação. A forma no qual seus átomos interagem é característica, porém as constantes da Equação 1, como A , por exemplo, se diferenciam, sendo necessário com isso o ajuste adequado das mesmas. A MCSE nos permite, em tempo real, alterar esses parâmetros que caracterizam unicamente cada um dos elementos citados acima, por exemplo. A Figura 2 apresenta a interface do programa, onde os campos A , J e D são editáveis e correspondem as constantes da Equação 1.

Conforme visto na Seção 2, a magnetização interna (ou local) de um sistema composto por elementos ferromagnéticos depende de fatores externos, como o campo e a temperatura no qual ele está submetido. A MCSE nos permite alterar de forma direta e simples esses dois fatores. O fator Kt (temperatura multiplicada pela Constante de Boltzmann) pode ser facilmente alterado de acordo com a nossa necessidade, como ilustra a Figura 2. O campo editável, chamado de Kt , recebe o valor que o usuário desejar. A mesma figura apresenta uma esfera utilizada no direcionamento do campo magnético externo; ao clicá-la e segurá-la, o vetor que representa o campo uniforme é girado simultaneamente, nos dando com isso total liberdade para alterar as condições que gostaríamos que fosse imposto ao sistema. Note que tanto as constantes quanto o campo externo podem ser alterados de forma simples e rápida.

O termo de longo alcance (interação dipolo-dipolo) da Equação 1 força-nos a calcular,

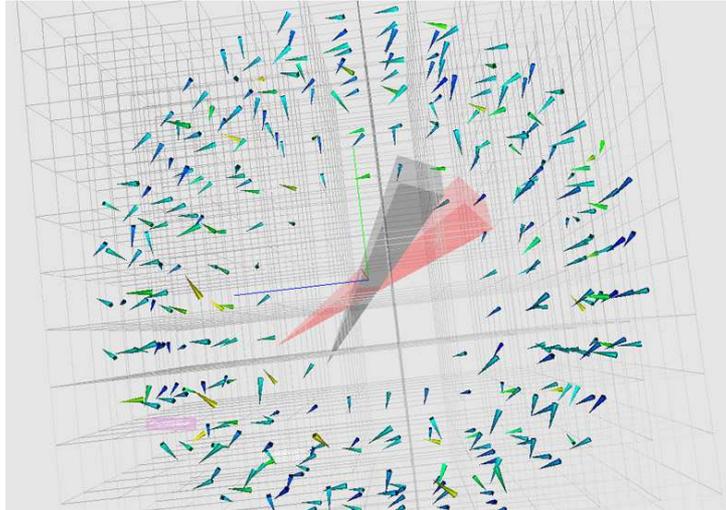


Figura 3: Representação dos *spins* dentro de uma grade de simulação regular, os dois vetores no centro mostram o campo externo e a magnetização do arranjo.

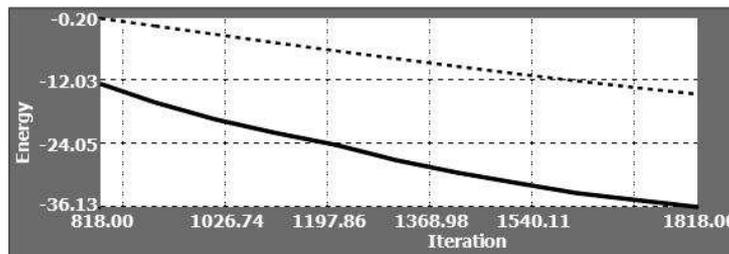


Figura 4: Gráfico que informa a média das energias e seu somatório total(tracejado)

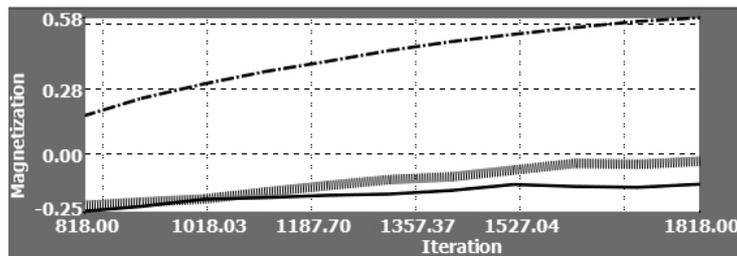


Figura 5: Desenvolvimento da magnetização ao longo do número de iterações concluídas. Cada linha indica como ela evolui pelos três eixos coordenados(x,y,z)

a cada iteração, as distâncias entre os *spins*. Assim, podemos concluir que a energia total E_t está diretamente ligada a forma no qual estes estão dispostos na grade. A geometria do objeto influencia diretamente em sua energia e conseqüentemente em sua magnetização, visto que a magnetização interna depende dos *spins* e o direcionamento destes é influenciado pela energia total. Como apresentado na Seção 3.2, a geometria do objeto é definida através da equação implícita da forma que queremos adotar. Desta forma, sendo necessário alterar o arranjo dos *spins*, basta apenas modificar a formulação implícita do modelo geométrico que achamos conveniente ao problema.

Ao longo da simulação os *spins* vão girando, e conseqüentemente a energia e a magnetização

vão se alterando. Na MCSE podemos acompanhar a evolução de cada resposta ao decorrer do tempo: gráficos justapostos informam ao usuário como o sistema está reagindo às condições que foram escolhidas. Na Figura 4 apresentamos o comportamento da energia pelo número de iterações. O tracejado é utilizado para informar a média de todas as energias calculadas e a linha sólida o somatório total destas ao longo do tempo. Um segundo gráfico (Fig. 5) informa ao usuário como a magnetização do arranjo está respondendo às freqüentes mudanças de direção dos *spins*. Durante toda a simulação, o usuário pode acompanhar a evolução dos *spins* durante a simulação, bem vetores representando o campo externo que está atuando no arranjo e a magnetização do arranjo 3.

4. Resultados Experimentais

Foram realizadas cinco simulações com um número distinto de átomos, porém mantendo fixa a geometria do objeto e as constantes da equação de energia (Fig. 6). O critério adotado para a parada do algoritmo foi o valor médio de energia do sistema: quando este se torna estável a simulação é encerrada. Com este nível de energia, o arranjo molecular apresentava características de magnetização nula e os *spins* se encontravam praticamente alinhados com o campo magnético externo.

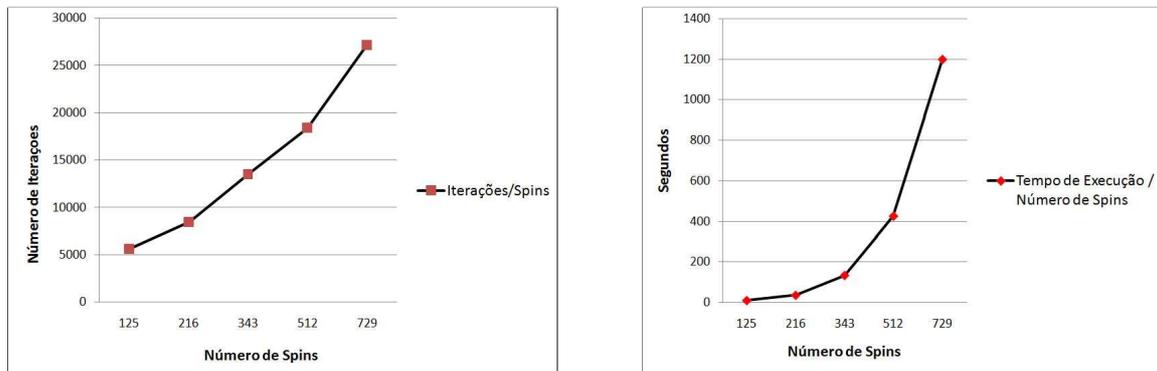


Figura 6: Variação de interações e do tempo de execução em função do número de átomos.

O objetivo da simulação foi analisar o comportamento do simulador, em termos de tempo de execução e número de iterações, diante da variação no número de spins de um arranjo. Os resultados mostram que, a medida que aumentamos o número de spins, o tempo de execução aumenta com tendência exponencial. Como pretendemos criar arranjos com quantidades maiores de spins, de forma a simular estruturas de maior porte, torna-se evidente a necessidade de paralelizarmos nosso simulador, de forma a tirar proveito das modernas arquiteturas paralelas providas pelas máquinas multi-núcleo e pelas GPUs de propósito geral. Esses resultados foram obtidos através da simulação feita em uma máquina Pentium Core 2 Quad Q6600 com 4GB de memória RAM e sistema operacional Windows Vista 64 bits Ultimate.

É importante salientar que o número de iterações não tem o mesmo significado que o tempo de execução. Como o objeto tem a mesma geometria mas número de *spins* diferente, é necessário um número diferente de iterações para que a energia média convirja para a energia de referência. Porém, o tempo de execução de cada iteração depende do volume no qual o objeto está definido, ou seja, do seu suporte em \mathbb{R}^3 .

5. Trabalhos Correlatos

Encontra-se na literatura trabalhos semelhantes ao proposto neste artigo. Diferenças na modelagem da parte física e estrutura computacional influenciam muito no resultado e no de-

sempenho do sistema. Como exemplo, pode-se considerar alguns artigos que abordam a parte do cálculo da energia de uma forma diferente, omitindo o termo de longo alcance (interação dipolo-dipolo) e usando apenas os termos ferromagnéticos e de campo externo para encontrar o ponto crítico do sistema (Landau & Binder, 2005; Leite & Figueiredo, 2007).

Como visto anteriormente, o termo dipolo-dipolo é o mais custoso entre os três em aspecto computacional, pois requer dados de todos os outros átomos do conjunto. Omitindo este termo, consegue-se um redução na complexidade computacional, passando de $O(n^3)$ para uma complexidade linear $O(n)$. O tempo de processamento cai consideravelmente, porém sua omissão acarreta em uma não confiabilidade por completa dos dados colhidos. Os avanços na parte de hardware, a criação de processadores multinucleados e o desenvolvimento de placas gráficas mais poderosas e com suporte a implementação direta, fornece uma gama de possibilidades ao desenvolvimento de aplicações com grande custo computacional, por esse e outros motivos, é proposta uma aplicação que requer um processamento mais intenso, porém com resultados mais condizentes e confiáveis.

6. Trabalhos Futuros

O custo computacional para o cálculo da interação dipolo-dipolo, cuja complexidade computacional é da ordem de $O(n^3)$, é hoje um fator limitante em nossa simulação. Em última instância, quanto maior a quantidade de *spins* a ser simulado, mais demorada será a simulação. Assim, nossos esforços futuros terão como foco a busca de alternativas que tornem computacionalmente viável simulações com uma quantidade maior de *spins*. Para alcançar este propósito utilizaremos técnicas de programação paralela. Em um primeiro momento, desenvolveremos uma versão *multithread* de nossa aplicação, que possa explorar o paralelismo existente nos recentes processadores multi-núcleo. Em um segundo momento, pretendemos explorar o grande número de núcleos de execução providos pelas placas gráficas de propósito geral (GPGPU) (Owens et al., 2008). Estas duas versões serão disponibilizadas na forma de módulos, que poderão ser selecionados pelos usuários através da interface gráfica. Por fim, pretendemos desenvolver um terceiro módulo para aprimorar a visualização do arranjo de *spins*.

Além de explorar o paralelismo existente no cálculo da interação dipolo-dipolo, a versão multithread também procurará separar o processamento de dados do processamento gráfico, a fim de melhorar o desempenho final do sistema. Pretendemos utilizaremos a biblioteca PThreads (Butenhof, 1997) para o desenvolvimento desta versão.

A versão que utilizará GPGPUs para o cálculo da interação dipolo-dipolo utilizará CUDA (NVIDIA, 2007) para explorar o alto grau de paralelismo oferecido por este hardware. Assim, as partes paralelizáveis do processamento serão enviadas a placa de vídeo, que realizará os cálculos e retornará os resultados.

O terceiro módulo consiste em reconstruir o arranjo de *spins* de modo real, utilizando para isso uma biblioteca de manipulação geométrica de malhas, a GCGPolygon. Este módulo terá como finalidade estender a simulação para estruturas mais complexas, como, por exemplo, os nanotubos. Os três tipos de nanotubos, *zigzag*, *chiral* e *armchair* (Kharissova et al., 2007), serão modelados e simulados, com o objetivo de se observar o comportamento dessas estruturas em diversas condições físicas.

7. Conclusão

A simulação de interação entre átomos de elementos e compostos ferromagnéticos são muito importantes em estudos de física da matéria condensada. A limitação em sistemas de simulação está na alta complexidade computacional de modelos matemáticos próximos da realidade. Isso implica que esses estudos necessitam de computação de alto desempenho que

frequentemente possuem alto custo. Este trabalho visa propor um modelo computacional: (1) explore as características numéricas de equações de interação (Eq. 1); (2) facilite a modelagem de objetos magnéticos; (3) use tecnologias emergentes de alto desempenho como GPGPU (CUDA) e processadores multinucleados.

Os resultados apresentados demonstram que a boa compreensão das equações de interação de *spins* é fundamental para um sistema integrado de modelagem e simulação de objetos magnéticos. Foi possível obter resultados que envolvem objetos com geometria complexa e com tempo de execução razoável. Nesse sentido, a perspectiva futura é bastante animadora. A introdução de versões paralelas ampliará o leque de aplicações. Além disso, será possível realizar comparações objetivas entre as diversas tecnologias envolvidas.

REFERÊNCIAS

- Butenhof, D. R., 1997. *Programming with POSIX Threads*. Addison-Wesley Reading.
- dos Santos Cabral Neto, J., 2004. *Estudo Sistemático das Propriedades Termodinâmicas e criticalidade de Filmes Finos e Super-Redes Magnéticas*. PhD thesis, Universidade Federal de São Carlos.
- Fan, Z., Qiu, F., Kaufman, A., & Yoakum-Stover, S., 2004. Gpu cluster for high performance computing. In *SC '04: Proceedings of the 2004 ACM/IEEE conference on Supercomputing*, pp. 47, Washington, DC, USA. IEEE Computer Society.
- Grzybowski, A., Gwózdź, E., & Bródka, A., 2000. Ewald summation of electrostatic interactions in molecular dynamics of a three-dimensional system with periodicity in two directions. *PHYSICAL REVIEW B*, vol. 61, n. 10.
- Kharissova, O. V., Rangel-Cardenas, J., & Castanon, M. G., 2007. Structural computing of hamada parameters for carbon nanotubes processed by microwave heating. In *ICICIC '07: Proceedings of the Second International Conference on Innovative Computing, Information and Control*, pp. 357, Washington, DC, USA. IEEE Computer Society.
- Landau, D. P. & Binder, K., 2005. *A Guide to Monte Carlo Simulations in Statistical Physics, Second Edition*. Cambridge University Press, 2 edition.
- Leite, V. S. & Figueiredo, W., 2007. Determinação de curvas de histerese. *Revista Brasileira de Ensino de Física*, vol. 29, pp. 53–56.
- Matsumoto, M. & Nishimura, T., 1998. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Trans. Model. Comput. Simul.*, vol. 8, n. 1, pp. 3–30.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E., 1953. Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, vol. 21, pp. 1087–1092.
- NVIDIA, 2007. Nvidia cuda programming guide. Technical report, NVIDIA Corporation.
- Owens, J. D., Houston, M., Luebke, D., Green, S., Stone, J. E., & Phillips, J. C., 2008. Gpu computing. *Proceedings of the IEEE*, vol. 96, n. 5, pp. 879–899.
- Ribeiro, G. A. P., 2000. As propriedades magnéticas da matéria: um primeiro contato. *Revista Brasileira de Ensino de Física*, vol. 22, n. 3.
- Rodrigues, C. I., Hardy, D., Stone, J. E., Schulten, K., & Hwu, W. W., 2008. Gpu acceleration of cutoff pair potentials of molecular modeling applications.
- Velho, L., Gomes, J., & de Figueiredo, L. H., 2002. *Implicit Objects in Computer Graphics*. Springer-Verlag, 1 edition.