

**UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE
CAMPUS DE NATAL
BACHARELADO EM CIÊNCIA E TECNOLOGIA**

RODRIGO DE LIMA SANTANA

**DESENVOLVIMENTO DE SOFTWARE (MagSimu) PARA
MODELAÇÃO DE SISTEMAS MAGNÉTICOS NANOESTRUTURADOS**

**NATAL
2018**

RODRIGO DE LIMA SANTANA

**DESENVOLVIMENTO DE SOFTWARE (MagSimu) PARA
MODELAÇÃO DE SISTEMAS MAGNÉTICOS NANOESTRUTURADOS**

Monografia apresentada ao Departamento de
Ciência e Tecnologia como requisito para a
obtenção do título de bacharel em Ciência e
Tecnologia.

Orientador: Prof. Dra. Ana Lúcia Dantas

**NATAL
2018**

RODRIGO DE LIMA SANTANA

**DESENVOLVIMENTO DE SOFTWARE (MagSimu) PARA MODELAÇÃO DE
SISTEMAS MAGNÉTICOS NANOESTRUTURADOS**

Monografia apresentada ao Departamento de
Ciência e Tecnologia como requisito para a
obtenção do título de bacharel em Ciência e
Tecnologia.

Aprovada em ___/___/___.

BANCA EXAMINADORA

Titulação. Nome e sobrenome - Orientador
Universidade do Estado do Rio Grande do Norte

Titulação. Nome e sobrenome
Universidade do Estado do Rio Grande do Norte

Titulação. Nome e sobrenome
Universidade do Estado do Rio Grande do Norte

Dedico este trabalho a todos os que me ajudaram e
contribuíram para a sua finalização, em especial a
minha orientadora Ana Lúcia Dantas.

AGRADECIMENTOS

A minha orientadora Prof. Dr. Ana Lúcia Dantas, que sempre teve disposição para me auxiliar quando me encontrava com problemas, tanto no projeto quanto na elaboração deste trabalho, e que confiou na minha capacidade e me motivou desde o início.

A minha família, que sempre se preocupou comigo e me apoiou de forma moral, emocional, financeira, e em tudo que eu precisava, não me deixando faltar nada.

Aos meus amigos e colegas de faculdade, Thalita Queiroz, Pâmala Samara, Alice Costa, Irlan Camelo, Natália Ramos e Tanara Inácio, que sempre estiveram do meu lado como uma segunda família e me deram um grande apoio em todas as minhas dificuldades. Sei que posso contar com eles sempre.

Aos colegas e amigos do grupo de magnetismo, que me tiraram muitas dúvidas e me ajudaram em vários problemas relacionados ao projeto que deu origem a este trabalho.

A CNPq, pela oportunidade que me deu e pelo apoio financeiro que me possibilitou seguir com este trabalho até o fim sem faltar nada.

A UERN, por sempre se mostrar atenciosa com os seus alunos e por estar sempre disposta a nos ajudar.

Sou verdadeiramente grato a todos que contribuíram de alguma forma com este trabalho, mesmo que não esteja ciente de suas contribuições.

RESUMO

O avanço de técnicas de crescimento e de caracterização em materiais da ordem de nanômetros fomentou o desenvolvimento de novos materiais, com novas propriedades e novas funcionalidades, dando origem ao que hoje chamamos de nanotecnologia. Com os aprimoramentos tecnológicos que houve com o passar dos anos, tornou-se possível fazer cálculos numéricos complexos muito rapidamente, o que levou pesquisadores a implementar programas que simulassem as propriedades de nanopartículas magnéticas. O problema é que muitos desses softwares, até hoje, apresentam uma clara complexidade de operação, uma vez que ou são rotinas que só são possíveis de ser operadas por quem entende de programação ou apenas apresentam grande dificuldade de usabilidade/instalação. Pensando nesse problema, este trabalho tem como objetivo apresentar um software de alto desempenho, desenvolvido em Java, que atenda a crescente demanda de físicos, engenheiros, farmacêuticos e até mesmo médicos, focando no desenvolvimento de uma interface amigável que viabilize o uso dos pesquisadores, que mesmo sem profundo conhecimento de programação, possam interpretar suas medidas magnéticas em sistemas magnéticos nanoestruturados. O software, que nós nomeamos MagSimu, é uma ferramenta de simulação magnética nanoestruturada que foi desenvolvida com base nos requisitos levantados pela equipe. Os requisitos são: ter uma interface gráfica simples e fácil de usar; calcular as configurações de equilíbrio de um sistema magnético, pelo método auto-consistente, com um campo variante externo; salvar os resultados do programa em arquivos que podem ser analisados no OriginLab 9.0 (ou qualquer outro que tenha o mesmo propósito); e tenha código aberto. O desenvolvimento do código-fonte do programa foi realizado do início ao fim através do NetBeans IDE 8.2, o que facilitou a configuração da interface gráfica, além de ter várias bibliotecas úteis para o desenvolvimento do programa. Após a conclusão do software MagSimu, foram realizados experimentos comparando os resultados do programa com os dados de uma fonte de maior autoria, o que, por sua vez, permitiu verificar a validade dos resultados gerados pelo programa. Por fim, tendo este e os demais requisitos sido atendidos satisfatoriamente, o MagSimu se mostrou capaz de ser implementado, de modo que é necessário apenas que se tenha conhecimento suficiente em nanomagnetismo para operá-lo, e não será essencial que se tenha contato com o código-fonte devido à interface, embora seja possível a um programador modificar o código para se adequar a um objetivo específico de pesquisa.

Palavras-chave: Nanomagnetismo. Sistemas magnéticos nanoestruturados. MagSimu. Método auto-consistente. Interface amigável.

ABSTRACT

The progress of increase and characterization techniques in materials of order of nanometers instigated the development of new materials, with new properties and functionalities that give rise to what we know today by nanotechnology. With the technological enhancements which there was recently, become possible to make complexes numerical calculations very quickly, which led researchers to implement programs that simulate the properties of magnetic nanoparticles. The problem is that many of these softwares, until today, presents an evident complexity of operation, since they are routines which are only possible to be operated by who knows programming or just presents great difficult to usability/setup. Looking at the problem, This paper aims to present a high-performance software, developed in Java, and considering the increasing requirement of experimentalists, physicists, engineers, pharmacists and even physicians, focusing on development of a friendly interface that to make feasible the use by researchers, who even without deep knowledge in programming, for interpreting their magnetic measurements in nanostructured magnetic systems. The software, which we nominated MagSimu, is a nanostructured magnetic simulation tool that was developed based on the claims raised by the team. The requirements are: have a simple and easy to use graphical user interface; calculate the equilibrium settings of a magnetic system, by the self-consistent method, with an external variant field; save the program results into files that can be parsed in OriginLab 9.0 (or any other that has the same purpose); and have open source. The development of the program's source code was carried out from start to finish through the NetBeans IDE 8.2, which made it easy to set up the graphical interface, as well as having several useful libraries for the development of the program. After completion of the MagSimu software, experiments were performed comparing the results of the program with the data from a higher-level source, which, in turn, made it possible to verify the validity of the results generated by the program. Finally, having this and the other requirements met satisfactorily, the MagSimu is capable of being implemented, so it is only necessary that one has enough knowledge in nanomagnetism to operate it, and it will not be essential to have contact with the source code due to the interface, although it is possible to a programmer modify the code to suit for a specific objective for research.

Keywords: Nanomagnetism; nanostructured magnetic systems; MagSimu; self-consistent method; friendly interface

LISTA DE ILUSTRAÇÕES

Figura 3.1 – Curva de magnetização	18
Figura 3.2 – Configurações de equilíbrio	18
Figura 4.1 – Perfil de magnetização de um material ferromagnético	20
Figura 4.2 – Exemplo de célula de simulação	21
Figura 5.1 – Tela inicial do programa MagSimu	26
Figura 5.2 – Programa operando com as configurações da Figura 5.1	26
Figura 5.3 – Compilação de programas em Java pela JVM	29
Figura 5.4 – Fluxograma do método numérico auto-consistente	32
Figura 5.5 – Diagrama de classes simplificado do software MagSimu	33
Figura 5.6 – Diagrama de atividade do software MagSimu	35
Figura 6.1 – Diagrama de fase de discos (prismas cilíndricos) de ferro e de permalloy baseado na influência do diâmetro e da espessura	37
Figura 6.2 – Ilustrando as fases magnéticas de um disco	38
Figura 6.3 – Tela com as informações que os experimentos terão em comum	39
Figura 6.4 – Gráfico de magnetização do disco de Ferro (75nm×10nm) com os mapas de vetores da primeira camada de células do disco para $H = 2$ kOe e $H = 0$ kOe	40
Figura 6.5 – Gráfico de magnetização do disco de Ferro (75nm×25nm) com os mapas de vetores da primeira camada de células do disco para (a) $H = 2$ kOe e (b) $H = 0$ kOe	41
Figura 6.6 – Gráfico de magnetização do disco de Permalloy (45nm×15nm) com os mapas de vetores da primeira camada de células do disco para (a) $H = 2$ kOe e (b) $H = 0$ kOe	42
Figura 6.7 – Gráfico de magnetização do disco de Permalloy (45nm×25nm) com os mapas de vetores da primeira camada de células do disco para (a) $H = 2$ kOe e (b) $H = 0$ kOe	44

Figura A.1 – Classe “Principal”	47
Figura A.2 – Classe “Entrada”	48
Figura A.3 – Classe “JRede”	49
Figura A.4 – Classe “CampoMagnético”	50
Figura A.4 – Classe “Resultados”	51
Figura A.6 – Classe “Retangular”	52
Figura A.7 – Classe “Eliptico”	52
Figura A.8 – Classe “PrismaRetangular”	53
Figura A.9 – Classe “PrismaEliptico”	53
Figura A.10 – Classe “Esfera”	54

SUMÁRIO

1. INTRODUÇÃO	11
2. MAGNETISMO E NANOESTRUTURAS MAGNÉTICAS	15
3. MATERIAIS FERROMAGNÉTICOS.....	17
4. ENERGIAS MAGNÉTICAS E CAMPOS MAGNÉTICOS	19
4.1. ENERGIA DE TROCA.....	22
4.2. ENERGIA ZEEMAN	22
4.3. ENERGIA DE ANISOTROPIA	22
4.4. ENERGIA DIPOLAR	23
4.5. CAMPO MAGNÉTICO LOCAL SOBRE CÉLULAS.....	23
4.5.1. Campo de troca	24
4.5.2. Campo Zeeman	24
4.5.3. Campo de anisotropia.....	24
4.5.4. Campo dipolar.....	25
5. ESTRUTURA E FUNCIONAMENTO DO SOFTWARE MagSimu	25
5.1. APRESENTANDO O MagSimu.....	25
5.2. TECNOLOGIA E MÉTODOS UTILIZADOS.....	29
5.2.1. Java.....	29
5.2.2. NetBeans IDE 8.2	30
5.2.3. Método Numérico Auto-Consistente	31
5.3. REQUISITOS DO MAGSIMU	32
5.4. ANÁLISE DOS DIAGRAMAS	32
5.4.1. Diagrama de classes	33
5.4.2. Diagrama de atividade.....	35
6. AVALIAÇÃO DOS RESULTADOS	36
6.1. EXPERIÊNCIA 1:.....	39
6.2. EXPERIÊNCIA 2:.....	40
6.3. EXPERIÊNCIA 3:.....	42
6.4. EXPERIÊNCIA 4:.....	43
7. CONCLUSÃO	45
REFERÊNCIAS	46
APÊNDICE A – Atributos e métodos das classes do MagSimu	47
A.1. CLASSE “Principal”	47
A.2. CLASSE “Entrada”	48
A.3. CLASSE “JRede”	49

A.4. CLASSE “CampoMagnetico”	49
A.5. CLASSE “Resultados”	50
A.6. CLASSE “Retangular”	51
A.7. CLASSE “Eliptico”	52
A.8. CLASSE “PrismaRetangular”	52
A.9. CLASSE “PrismaEliptico”	53
A.10. CLASSE “Esfera”	53

1. INTRODUÇÃO

O avanço de técnicas de crescimento e de caracterização em materiais da ordem de nanômetros (10^{-9} m), fomentou o desenvolvimento de novos materiais, com novas propriedades e novas funcionalidades dando origem ao que hoje chamamos de nanotecnologia.

A nanotecnologia promoveu grandes avanços científicos em grandes áreas do conhecimento, como por exemplo, nas áreas de química, bioquímica, biologia, farmácia, saúde, física, etc. Do ponto de vista de magnetismo, área multidisciplinar, revolucionou a área de gravação magnética dando origem a uma nova área de conhecimento chamada de spintrônica, bem como ampliou, significativamente, seu potencial de aplicações biomédicas.

O desenvolvimento de novos sensores magnéticos, como por exemplo, o cabeçote de leitura magnetorresistivo, foi iniciado graças a experimentos como os descritos no trabalho de Baibich (BAIBICH, 1988). Baibich e colaboradores descobriram a magnetorresistência de sistemas artificiais, construídos a partir da justaposição de filmes ferromagnéticos, de ferro (Fe) e cromo (Cr), em multicamadas. O trabalho original comprovou variações expressivas da resistência elétrica, devido a mudanças nas fases magnéticas das multicamadas Fe/Cr em campos magnéticos externos com intensidade da ordem de alguns kOe.

Devido ao seu potencial tecnológico, o sistema logo despertou o interesse da indústria. E já em 1990, o grupo de Stuart Parkin da IBM desenvolveu sistemas semelhantes com fraco acoplamento entre os filmes magnéticos (DIENY, 1990), com variação da resistência elétrica em campos externos com intensidade da ordem de poucos Oe. A proposta de Parkin abriu um novo cenário para a inovação tecnológica na indústria de gravação magnética.

A partir da proposta de Parkin foi desenvolvido o dispositivo conhecido como válvula de spin que revolucionou toda a indústria de gravação magnética, dando origem a várias frentes de pesquisa que buscavam atender as demandas da sociedade por aumentar cada vez mais a densidade de gravação magnética. Vale registrar que o dispositivo válvula de spin está para a spintrônica, assim como o transistor está para a eletrônica.

Com a disponibilidade de tecnologia de alta qualidade para o crescimento de nanopartículas magnéticas, uma das frentes de pesquisa que mais se expandiu nessa área foi a de desenvolvimento de sistemas magnéticos nanoestruturados (*dots* ou nanopartículas, nanofios, filmes finos, nanodiscos, multicamadas, etc.)

Todavia, essa busca pela miniaturização dos sistemas magnéticos do ponto de vista experimental trouxe consideráveis dificuldades para os modelos teóricos existentes, em sua maioria analíticos, como o modelo Stoner-Wohlfarth (STONER e WOHLFARTH, 1948), que

conta com um mecanismo uniforme de reversão da magnetização. O modelo de Stoner-Wolfarth tem limitação para descrever sistemas magnéticos em que as superfícies são parte relevante do sistema. O problema da miniaturização é quanto menor for o sistema mais importante é sua superfície, e nesse caso a energia dipolar passa a ser parte relevante da energia magnética.

Não é simples trabalhar, teoricamente, com a energia dipolar. A energia dipolar é a energia magnética derivada da interação entre todos os momentos magnéticos do sistema. Como, para cada átomo magnético das nanopartículas, existe um momento magnético associado, o número de variáveis a ser considerado aumenta muito. Portanto, é impossível se trabalhar com modelos analíticos e até mesmos os modelos numéricos são limitados pela capacidade de processamento de cálculo dos computadores.

Sendo assim, é importante desenvolver estratégias numéricas para diminuir a grande quantidade de variáveis, sem comprometer a descrição das características magnéticas do sistema nanoestruturado. Nesse contexto, uma proposta aceita e consolidada pela comunidade científica é a teoria micromagnética.

A teoria micromagnética, embora o termo “micro” se associe a dimensão de 10^3 nm, se adequa bem a sistemas magnéticos nanoestruturados nas dimensões de 1 nm a 1000 nm. Segundo Aharoni (1996), esta teoria foi desenvolvida para “negligenciar” a mecânica quântica e ignorar a natureza atômica da matéria para usar a física clássica em um meio contínuo de modo a garantir valores muito próximos da realidade.

Como artifício para diminuir o número de variáveis numéricas do sistema, a teoria micromagnética introduz o conceito de célula de simulação numérica que representa uma pequena região cúbica, com dimensões d^3 (d da ordem de alguns nanômetros), em que todos os momentos magnéticos se orientam, em média, na mesma direção e são associados a um momento magnético médio.

Todavia, mesmo com o artifício da teoria micromagnética, considerando que a energia de interação dipolar é de longo alcance, a modelagem numérica de um sistema de algumas dezenas de nanômetros pode ser inviabilizada em computadores que não possuam alto desempenho de processamento. A energia dipolar, também conhecida como energia de forma, introduz uma dependência, relativamente forte, das propriedades magnéticas na geometria dos sistemas magnéticos nanoestruturados. Isso implica que mesmo os computadores pessoais que possam lidar com nanoestruturas simples e muito pequenas, ao aumentar as suas dimensões, o processo pode se tornar lento ou até inviável para as capacidades da máquina. Dada essa limitação, é altamente recomendado que tais processos ocorram em computadores de médio a

grande porte, pois possuem capacidade de processamento elevado, que comporta esses processos.

Nosso grupo de pesquisa tem experiência consolidada na área de modelagem de sistemas magnéticos nanoestruturados. Todavia, o grupo sempre trabalhou com linguagem de programação estruturada que, embora seja de código aberto, dificilmente dispõe de uma interface amigável para o usuário que não seja especialista em programação.

Portanto, considerando a demanda cada vez maior dos experimentais, físicos, engenheiros, farmacêuticos e até médicos, por interpretação de suas medidas magnéticas em sistemas magnéticos nanoestruturados, apresentamos nesse trabalho um software de alta performance, em linguagem orientada a objeto, focando no desenvolvimento de uma interface amigável, de modo a viabilizar que pesquisadores, mesmo sem profundo conhecimento de programação possam calcular configurações de equilíbrio magnético, bem como magnetização, susceptibilidade, entre outras grandezas magnéticas para algumas geometrias específicas.

O diferencial de nosso software consiste em criar uma interface amigável e de código aberto de programação que possa analisar os efeitos geométricos, bem como os efeitos de campo magnético externo aplicado nas configurações de equilíbrio. Futuramente, efeitos de temperatura, e/ou efeitos de corrente polarizada de spin poderão ser incorporados de maneira relativamente simples ao código de programação.

Esse é um projeto experimental que se propõe a estudar as configurações de equilíbrio magnéticos de sistemas nanoestruturados, onde a energia magnética é composta pelas energias de troca, de anisotropia, dipolar e Zeeman. Os materiais magnéticos considerados são o ferro (Fe), permalloy, que consiste de uma liga de $(\text{Ni}_{80}\text{Fe}_{20})$, níquel (Ni) e cobalto (Co) nas geometrias retangulares, cilíndricas e esféricas.

A configuração de equilíbrio de um sistema magnético ocorre quando este se encontra em um estado mínimo de energia magnética. Ela é calculada a partir da teoria micromagnética, baseada no método de campo local auto-consistente. Uma descrição detalhada da metodologia usada nos modelos teóricos dos programas de pesquisa pode ser encontrada nas referências (DANTAS et al., 2005; DANTAS et al., 2007; DANTAS et al., 2009; REBOUÇAS et al., 2009; DANTAS et al., 2010; SILVA et al., 2011) para sistemas ferromagnéticos em que a interação dipolar de longo alcance é considerada.

Existem vários exemplos de programas já desenvolvidos com a mesma finalidade que o programa em questão. Dentre eles, há um serviço disponibilizado para o público na página do NIST (<http://math.nist.gov/oommf/>) que oferece uma rotina (Object Oriented Micro Magnetic

Framework) de código fechado para esse fim, que é usado pela grande maioria dos que investigam propriedades de sistemas magnéticos de dimensões sub-micrométrica.

Todavia, essa não é uma plataforma simples de ser usada/instalada, nem nos permite fazer alterações do código para que outras geometrias possam ser contempladas. Para mudarmos isso, estamos desenvolvendo esse software para que possamos disponibilizar uma biblioteca com maior diversidade de geometrias e materiais nanoestruturados, que podem ser operados por meio de uma interface gráfica amigável. Isso eliminará a necessidade de conhecer a linguagem por trás do programa, exceto na situação de o usuário precisar modificar algo ou acrescentar recursos ao programa, a depender dos interesses de pesquisa.

O código deste software foi implementado em Java, uma linguagem orientada a objetos, através do NetBeans 8.2, que possui um conjunto de ferramentas para criação e execução de aplicações Java e inclui o compilador, as bibliotecas, a máquina virtual (JMV) e outros utilitários. O programa também gera arquivos com os resultados do processo que podem ser analisados por meio do software Originlab 9.0, que pode gerar curvas numéricas, além de mapas de vetores.

Este trabalho está organizado em 4 partes: introdução, referencial teórico, materiais e métodos, e considerações finais. Nesta introdução temos a ideia do trabalho sendo apresentada e contextualizada, os objetivos a serem cumpridos, a justificativa e metodologia usada para esta pesquisa. No referencial teórico, explicamos os assuntos relacionados à área que este trabalho está envolvido. Assim mostramos a definição dos conteúdos importantes à pesquisa para que sejam esclarecidas.

Em materiais e métodos, serão descritos os passos da pesquisa, apresentaremos os aspectos e recursos do software desenvolvido e a sua forma de funcionamento, além de explicar os métodos utilizados para os cálculos e geração de resultados. Alguns testes serão feitos para mostrar a integridade e confiabilidade do programa. Nas considerações finais, discutiremos e verificaremos os impactos que este programa pode proporcionar.

2. MAGNETISMO E NANOESTRUTURAS MAGNÉTICAS

Os estudos sobre magnetismo evoluíram muito nas últimas décadas, o que fez esta se tornar uma das áreas de maior importância para o mundo moderno. Desde a invenção da bússola até o armazenamento de grandes quantidades de informação através da gravação magnética, o magnetismo se provou ser bastante útil de lá para cá.

A partir do desenvolvimento da nanotecnologia, vários ramos da ciência se propuseram a estudar os efeitos da miniaturização. Na área de magnetismo, não foi diferente.

A pesquisa de materiais magnéticos nanoestruturados está ganhando muita atenção hoje em dia, pois continua se abrindo uma alta gama de aplicações tecnológicas, e várias delas já fazem parte do cotidiano de todos. Aplicações estas como a gravação magnética em sistemas de alta densidade, cabeçotes de leitura e aplicações na biomédica como tratamento de tumores e câncer. Algumas dessas tecnologias, que ninguém imaginava serem possíveis, foram desenvolvidas graças a pesquisa de vários cientistas. Como diz Silva (2011, p.4) apud Baibich (1988), “tal tecnologia denominada por spintrônica teve início com a primeira observação do efeito magnetorresistivo gigante observado, em 1988, pelo físico Mário N. Baibich, quando trabalhava no laboratório do professor Albert Fert, prêmio Nobel de 2007”.

Uma das aplicações na área da biomédica é chamada de vetorização de fármacos, a qual injeta um ferrofluido no local da lesão, que possui partículas magnéticas que levam o medicamento acoplado a elas. A seguir, com o auxílio de um campo magnético externo, as partículas são magnetizadas e levadas até o local da lesão pelos vasos sanguíneos, se depositando no tecido doente e liberando o medicamento. Com relação a essa aplicação médica, para Dantas (2008, p.3):

Nesses sistemas é importante que as partículas sejam superparamagnéticas de modo a evitar danos a células nobres do sangue, durante o transporte para o alvo. As partículas se magnetizam em presença de campo externo, adquirem mobilidade magnetoforética, em presença de campo magnético e, portanto, podem ser utilizadas com sucesso para transportar princípios ativos que estejam ligados a elas, para separação celular e para se localizar em tumores permitindo o aquecimento local do tecido com a lesão.

Um dos principais desafios desta área está na gravação magnética. Desafio este que consiste em produzir sistemas magnéticos com densidades de gravação cada vez mais altas, o que significa que necessitamos de partículas ainda menores e bem arrançadas, além de espaços cada vez mais pequenos. Silva (2011, p.1) diz:

Um fator relevante neste aumento da capacidade de armazenar dados é o avanço das técnicas de construção de estruturas magnéticas cada vez menores, conduzindo a um relevante esforço de pesquisa em materiais magnéticos nanoestruturados, enfocando as pesquisas no seu comportamento, face a mudança de geometrias, campo de interface (estabilizante) entre outros fatores importantes.

O maior desafio quanto a miniaturização de sistemas magnéticos está no fato de que é impossível diminuir o tamanho das partículas indefinidamente, pois há um limite. Os cientistas estão sempre buscando o controle desse limite através do volume e da anisotropia dos materiais. Como explica Dantas (2008, p.3):

A barreira de energia que estabiliza partículas magnéticas é KV , onde K é a constante de anisotropia do material e V é o volume da nanopartícula. A estabilidade da ordem magnética requer $KV > k_B T$ (k_B é a constante de Boltzmann e T é a temperatura). Uma forma de manter a barreira de energia, diminuindo o volume da partícula, é aumentar a anisotropia. A dificuldade no uso de materiais de alta anisotropia é a necessidade de altos campos magnéticos para gravação, consequentemente altas correntes nos equipamentos eletrônicos.

A busca pela maior densidade de gravação e estabilidade das nanopartículas, fez os cientistas pesquisarem diversas maneiras para se alcançar este resultado. Há alguns métodos que ainda estão sendo pesquisados, que são: a gravação termicamente assistida e a matriz de dots. No primeiro, o objetivo é viabilizar o uso de materiais com altas anisotropias através da gravação com o auxílio um de laser, de modo que se obtenha bits cada vez menores (por este método, é possível gravar 100 vezes mais dados em uma mesma área). No segundo método, busca-se utilizar matrizes de *dots* (pontos magnéticos) para armazenar as informações em regiões muito pequenas, onde cada *dot* irá corresponder a um bit.

Existem vários grupos de pesquisa no campo do magnetismo no Brasil e no mundo. No Rio Grande do Norte, existe o grupo constituído por pesquisadores da Universidade do Estado do Rio Grande do Norte (UERN) e da Universidade Federal do Rio Grande do Norte (UFRN), e todos direcionaram seus focos para o estudo de sistemas magnéticos nanométricos, e isto inclui: (a) novos estados magnéticos de pontos, de fios, de filmes finos e de multicamadas magnéticas com impacto em aplicações de magneto-eletrônica e refrigeração magnética; e (b) produção de sistemas magnéticos nanoestruturados e desenvolvimento de aplicações biomédicas.

3. MATERIAIS FERROMAGNÉTICOS

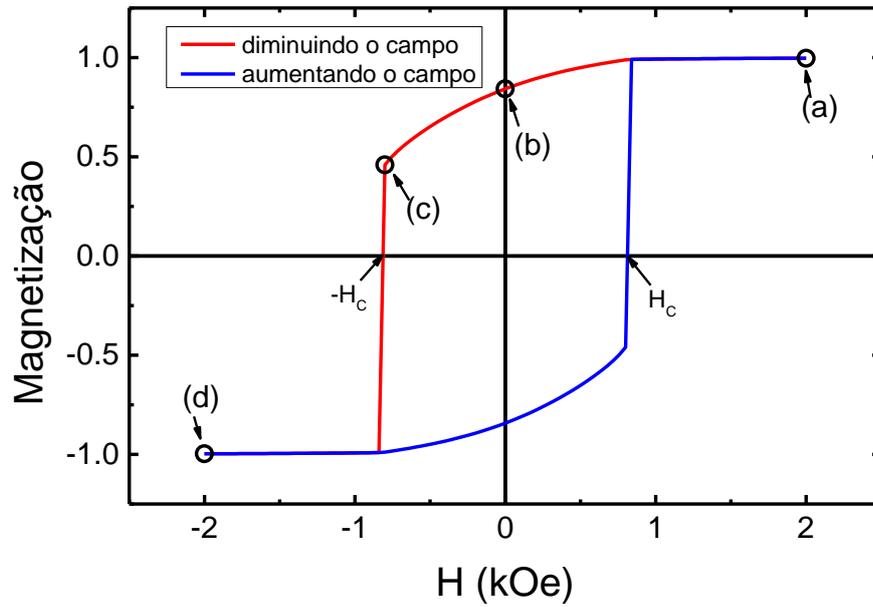
Segundo Frazão (2014, p. 7), os materiais ferromagnéticos se diferenciam dos demais por possuírem momentos magnéticos intrínsecos a sua estrutura que se ordenam espontaneamente ou se alinham facilmente mesmo sem influência de campo externo. Isso acontece porque a influência dos momentos magnéticos vizinhos é mais forte nesses materiais.

Esses materiais se magnetizam fortemente quando colocados na presença de um campo magnético. É possível verificar, experimentalmente, que a presença de um material ferromagnético altera fortemente o valor da intensidade do campo magnético (AGRA, 2015). Continuando Agra (2015):

São substâncias ferromagnéticas o Ferro (Fe), o Cobalto (Co), o Níquel (Ni) e algumas terras raras. Porém, uma vasta quantidade de ligas magnéticas podem ser formadas baseadas nessas substâncias e continuam apresentando propriedades magnéticas. Dentre estas ligas, duas delas se destacam: o Permalloy (Py) formado pela liga de $\text{Ni}_{80}\text{Fe}_{20}$ e a liga de $\text{Fe}_{73,5}\text{Cu}_1\text{Nb}_3\text{Si}_{13,5}\text{B}_9$ que será denominada sem os índices (FeCuNbSiB) [...], por apresentarem características magnéticas que otimizam as propriedades da dinâmica da magnetização [...].

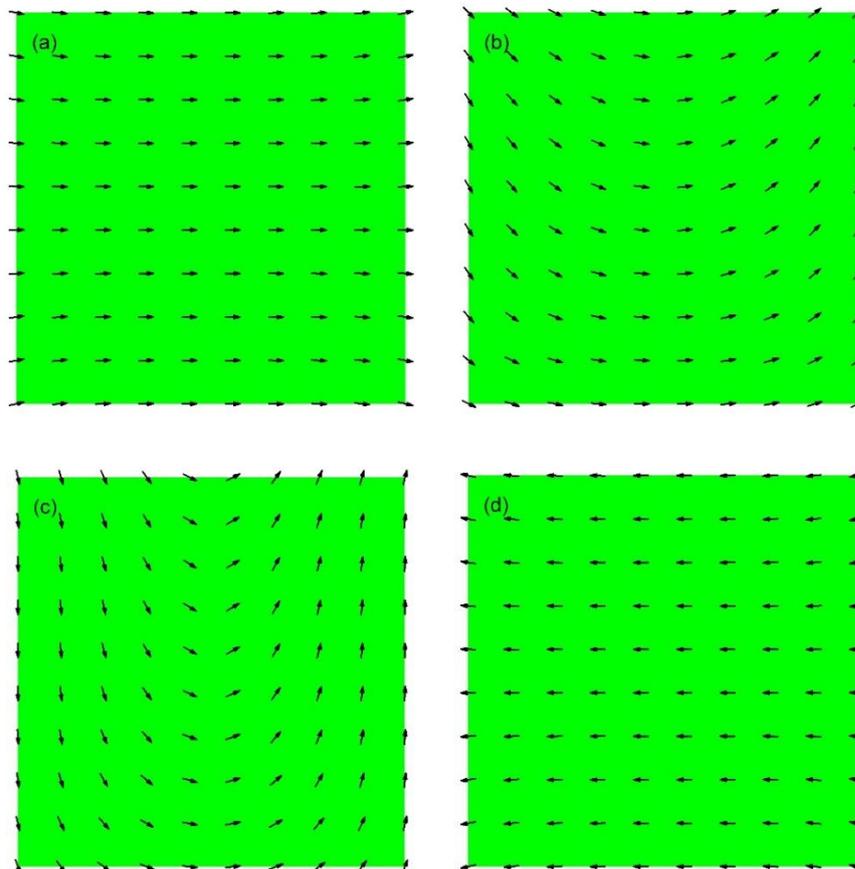
Uma medida experimental típica que identifica um material ferromagnético é a curva de histerese magnética (Figura 3.1). Na curva de histerese magnética alguns pontos são característicos e merecem destaques: o ponto (a) na Figura 3.1 mostra a saturação da magnetização, o que significa que todos os momentos magnéticos estão alinhados com o campo externo (ver Figura 3.2.a); o ponto (b) na Figura 3.1 mostra a magnetização remanente, que mostra que há magnetização residual mesmo quando não existe campo externo aplicado ($H = 0$) (ver Figura 3.2.b); o campo coercivo (H_C) é o valor do campo em que a magnetização é nula.

Figura 3.1 – Curva de magnetização.



Fonte: Elaborado pelo autor.

Figura 3.2 – Configurações de equilíbrio. (a) Saturação da magnetização em campo positivo. (b) Magnetização remanente. (c) Perfil de magnetização em $H = -0.8$ kOe, (d) Saturação da magnetização em campo negativo.



Fonte: Elaborado pelo autor.

Esse padrão de curva de histerese é a identidade mais simplificada de um material ferromagnético. Em nossos estudos nos concentraremos nos materiais de Co, Ni₈₀Fe₂₀, Fe e Ni. A largura da histerese, mede o campo coercivo $H_C = \frac{2K}{M_S}$, onde K é a constante de anisotropia e M_S é magnetização de saturação do material. Assim, a largura da histerese informa, de certa forma, sobre a anisotropia do material. Isto se as dimensões do material forem acima de micrometros. Se as dimensões do material forem nanométricos a energia de interação dipolar passa a ser parte relevante da energia magnética do sistema, e a largura pode ser alterada também pela energia de interação dipolar.

4. ENERGIAS MAGNÉTICAS E CAMPOS MAGNÉTICOS

Para descrever as configurações de equilíbrio de qualquer sistema físico é necessário identificar as energias que estão envolvidas e obter o mínimo da energia potencial. No caso de sistemas magnéticos, é importante identificar quais são as energias magnéticas que descrevem o sistema.

Nosso objetivo é o desenvolvimento de um software, com interface de usuários amigável, que possa descrever as fases magnéticas de equilíbrio e o processo de magnetização, por efeito de um campo magnético aplicado, em sistemas magnéticos nanoestruturados. Nesse caso, as energias relevantes são: energia de troca, energia de anisotropia, energia de interação dipolar e energia Zeeman.

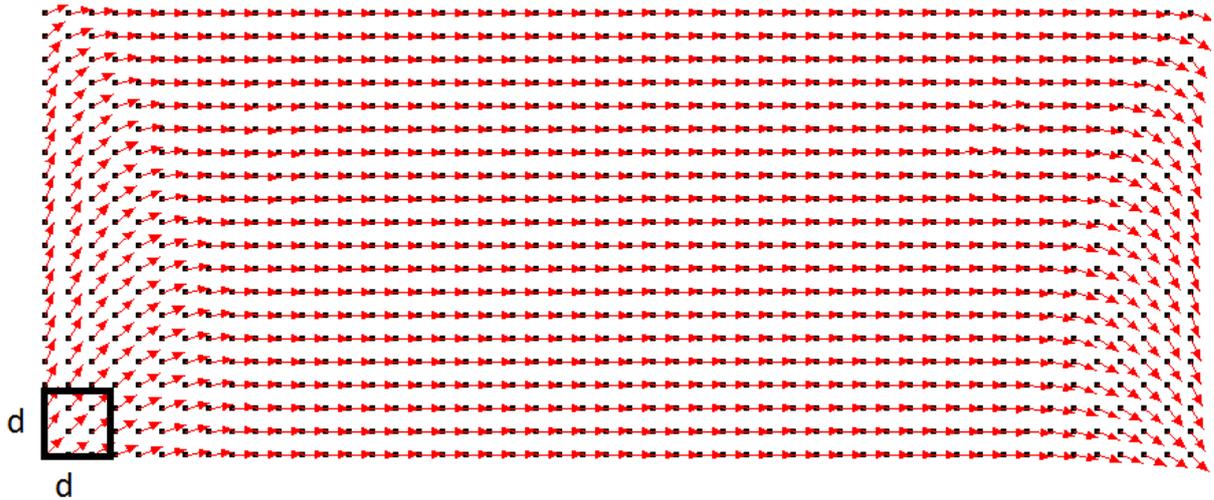
Como a energia de interação dipolar é de longo alcance, envolve a interação de cada momento magnético com todos os outros, a simulação numérica desse tipo de sistema físico requer estratégias numéricas, respeitando as propriedades intrínsecas do ponto de vista da física, para diminuir o número de variáveis.

No caso de sistemas magnéticos nanoestruturados, uma estratégia utilizada para diminuir o número de variáveis, tem sido a teoria micromagnética. A teoria micromagnética consiste em representar os momentos magnéticos associados a uma certa quantidade volumétrica do sistema, denominada de célula cúbica com dimensões d^3 , como um momento magnético efetivo por unidade de volume (M_S , magnetização de saturação).

Para o cálculo numérico computacional, as células são chamadas de células de simulação. Suas dimensões devem ser tais que a magnetização não apresente variação em seu volume. Como por exemplo, podemos ver na Figura 4.1, no caso bidimensional, como aplicar o conceito básico da teoria micromagnético de modo a diminuir o número de variáveis do sistema. Nesse exemplo, mostramos que a célula de simulação (quadrado preto, na Figura 4.1)

com dimensões quadradas de d^2 poderia representar um aglomerado de 3 x 3 momentos magnéticos. Em outras palavras estaríamos representando 9 momentos magnéticos em apenas 1, que é o momento magnético resultante da célula de simulação.

Figura 4.1 – Perfil de magnetização de um material ferromagnético. Em destaque a célula de simulação com dimensões quadradas de d^2 .



Fonte: Elaborado pelo autor.

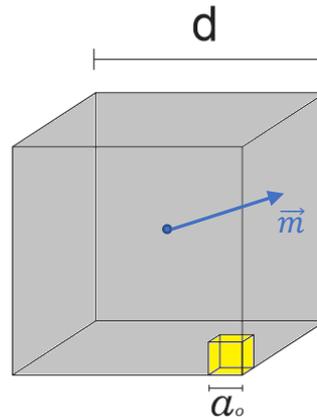
Se considerarmos uma estrutura cristalina do tipo BCC (cúbica de corpo centrado), com um volume d^3 (veja Figura 4.2). Se quisermos saber o número de átomos N que a estrutura possui, então:

$$N = 2 \left(\frac{d^3}{a_o^3} \right)$$

onde a_o é o parâmetro de rede do material, ou seja, a medida da aresta de suas células unitárias BCC. O resultado da fração é a quantidade de células unitárias BCC, e uma vez que este tipo de célula tem a capacidade de conter dentro de si dois átomos, se multiplicarmos a fração por 2, obtemos o número total de átomos da estrutura. Se calcularmos o número de átomos de um cubo de ferro com arestas $d = 20$ nm e parâmetro de rede $a_o^{\text{Fe}} = 0,287$ nm, temos:

$$N = 2 \left(\frac{20^3}{0,287^3} \right) = 2 \left(\frac{8000}{0,2364} \right) = 6,768 \times 10^5 \text{ átomos de ferro}$$

Figura 4.2 – Exemplo de célula de simulação.



Fonte: Elaborado pelo autor.

Se considerar esses $6,768 \times 10^5$ átomos no cálculo, tornar-se-ia realmente inviável calcular tantas variáveis, mesmo para um computador de alta potência.

As células de simulação devem ter um tamanho tal que a região a qual representa tenha todos os seus átomos com os momentos magnéticos aproximadamente uniformes. Se a estrutura magnética possui um tamanho com dimensões de centenas de nanômetros, torna-se mais viável usar o conceito de célula de simulação para realizar os cálculos. “A célula de simulação representa um volume do material magnético no qual não há mudanças consideráveis nos momentos magnéticos” (SILVA, 2014, p.32).

“O parâmetro escolhido para indicar o valor máximo da célula é o comprimento de troca” (REBOUÇAS, 2010, p.18). Segundo Rebouças (2010, p.18) apud Bertotti (1998), o comprimento característico que pode ser definido a partir de uma equação é chamado comprimento de intercâmbio ou *Exchange Length* (l_{EX}) e este parâmetro “decide” se uma certa porção de material é “pequena” ou “grande”. De acordo com ele, as porções consideradas “pequenas” tendem a magnetização uniforme, enquanto que as “grandes” tendem a magnetização não uniforme.

As energias magnéticas são intrínsecas aos átomos, assim como também às células de simulação. Como foi mencionado anteriormente, as estruturas magnéticas a serem analisadas constituem-se de pequenas células, que têm magnetização uniforme ou muito próxima disso, mas sem interferir na precisão dos resultados, e então possuindo quase as mesmas características dos átomos no cálculo. Cada célula tem uma energia efetiva a que é associada,

esta é dividida em alguns tipos de energia magnética que possuem características diferentes uma da outra. Iremos tratar de cada tipo de energia com mais detalhes nos próximos tópicos.

4.1. ENERGIA DE TROCA

A energia de interação de troca é uma energia de origem Coulombiana restrita a interação de primeiros vizinhos. Na teoria micromagnética, uma célula cúbica i , com dimensão d^3 e momento magnético $M_s \hat{m}_i$, interage exclusivamente com os momentos magnéticos das células primeiras-vizinhas. Nesse caso, a energia de troca por unidade de volume é dada por:

$$E_{Troca} = \frac{A}{d^2} \sum_{i,k} [1 - \hat{m}_i \cdot \hat{m}_k]$$

onde o parâmetro A é constante de troca entre os momentos magnéticos das células primeiras-vizinhas e \hat{m}_i é a direção do momento magnético da célula i . A soma em i representa a soma sobre todas as células do sistema e a soma j é a soma sobre todas as primeiras células vizinhas de cada célula i . A energia de troca em um material ferromagnético é mínima quando os momentos magnéticos são paralelos.

4.2. ENERGIA ZEEMAN

A energia Zeeman é a energia de interação entre o momento magnético das células do material e o campo magnético externo aplicado. Pode ser calculado da seguinte forma:

$$E_{Zeeman} = -\vec{H} \cdot M_s \sum_i \hat{m}_i$$

onde M_s é o momento de spin e \vec{H} é o campo magnético externo. Essa energia apresenta intensidade mínima quando o momento magnético e o campo magnético externo estão alinhados na mesma direção e sentido.

4.3. ENERGIA DE ANISOTROPIA

A energia de anisotropia é, segundo Guimarães (2009, p.32), a energia de uma amostra magneticamente ordenada que depende da direção relativa da magnetização e dos eixos estruturais; no caso mais simples, anisotropia uniaxial o sólido tem um eixo ao longo do qual a

energia está no mínimo. Considerando a anisotropia uniaxial, eixo x , a anisotropia magnetocristalina é dada por:

$$E_{Anis} = -K \sum_i (\hat{m}_i^x)^2$$

onde K é a constante magnetocristalina.

4.4. ENERGIA DIPOLAR

A Energia Dipolar é a energia de interação entre os momentos magnéticos de todas as células. Também conhecida como energia magnetostática e pode ser dada por:

$$E_{Dipolar} = \frac{M_S^2}{2} \sum_{i,k} \left(\frac{\hat{m}_i \cdot \hat{m}_k}{r_{ik}^3} - \frac{3(\hat{m}_i \cdot \hat{m}_k)(\hat{m}_k \cdot \hat{r}_{ik})}{r_{ik}^5} \right)$$

onde \vec{r}_{ik} é o vetor distância entre os momentos magnéticos das células i e k ; i representa a localização das células unitárias na rede do sistema e admite todas as posições, e k também admite todas as posições do sistema. A equação é dividida por 2 para corrigir a somatória, e assim evitar contagens duplas das energias do sistema.

4.5. CAMPO MAGNÉTICO LOCAL SOBRE CÉLULAS

Como vimos anteriormente a densidade energia magnética total é dada por:

$$E = \frac{A}{d^2} \sum_{i,k} [1 - \hat{m}_i \cdot \hat{m}_k] - \vec{H} \cdot M_S \sum_i \hat{m}_i - K \sum_i (\hat{m}_i^x)^2 + \frac{M_S^2}{2} \sum_{i,k} \left(\frac{\hat{m}_i \cdot \hat{m}_k}{r_{ik}^3} - \frac{3(\hat{m}_i \cdot \hat{m}_k)(\hat{m}_k \cdot \hat{r}_{ik})}{r_{ik}^5} \right)$$

O campo magnético efetivo, \vec{H}_{eff}^i , sobre cada célula i é definido a partir do gradiente da energia magnética com relação ao momento magnético da célula que é dado por:

$$\vec{H}_{eff}^i = -\vec{\nabla}_{m_i} E$$

A unidade de campo magnético H é Ampère por metro (A/m) no sistema internacional. Apesar disso, ainda é muito frequente utilizarem a unidade de medida Oersted (Oe) do sistema CGS para a medição de campo. Oersted também será utilizado para a medição de campo neste trabalho. Segue abaixo a equivalências de ambas as medidas.

$$4\pi \times 10^{-3} Oe = 1 A/m$$

4.5.1. Campo de troca

O campo de troca que atua em uma célula i produzido por suas células vizinhas pode ser obtido a partir da expressão da energia de troca e será dado por:

$$\vec{H}_{Troca}^i = -\frac{A}{M_S d^2} \sum_k \hat{m}_k$$

onde j corresponde as células de primeiros vizinhos das células i .

4.5.2. Campo Zeeman

O campo Zeeman é o campo externo aplicado à estrutura e se mostra com mesma intensidade em todas as células do sistema. A partir da energia Zeeman podemos obter o campo Zeeman, assim sendo dado por:

$$\vec{H}_{i,Zeeman}^p = \vec{H}$$

4.5.3. Campo de anisotropia

O campo de anisotropia uniaxial tem os valores máximos para a magnetização da célula sobre o eixo de energia mínima, que se alinham tentando buscar equilíbrio. O campo de anisotropia para cada célula de simulação será dado por:

$$\vec{H}_{Anis}^i = \frac{2K}{M_S} m_i^x \hat{x}$$

Nesse estudo, foi considerado o eixo x como o eixo fácil de magnetização.

4.5.4. Campo dipolar

Este é um campo de longo alcance. Cada célula i atua sobre todas as outras células k do sistema, e sua direção e sentido não é tão trivial quanto as direções e sentidos dos campos vistos anteriormente. Neste caso, escreveremos todas as componentes do campo dipolar. Calculando a componente x do campo dipolar, temos:

Na direção x :

$$H_{i,Dipolar}^x = \frac{M_S}{2} \sum_k \left(\frac{3r_{ik}^x (m_k^x r_{ik}^x + m_k^y r_{ik}^y + m_k^z r_{ik}^z)}{r_{ik}^5} - \frac{m_k^x}{r_{ik}^3} \right)$$

Na direção y :

$$H_{i,Dipolar}^y = \frac{M_S}{2} \sum_k \left(\frac{3r_{ik}^y (m_k^x r_{ik}^x + m_k^y r_{ik}^y + m_k^z r_{ik}^z)}{r_{ik}^5} - \frac{m_k^y}{r_{ik}^3} \right)$$

Na direção z :

$$H_{i,Dipolar}^z = \frac{M_S}{2} \sum_k \left(\frac{3r_{ik}^z (m_k^x r_{ik}^x + m_k^y r_{ik}^y + m_k^z r_{ik}^z)}{r_{ik}^5} - \frac{m_k^z}{r_{ik}^3} \right)$$

Agora que toda a teoria foi abordada, apresentaremos na próxima seção o software desenvolvido, que nomeamos de MagSimu, as suas funcionalidades e como utiliza do conhecimento mostrado até aqui, além da seção de Resultados que avalia, por meio de experimentos, a confiabilidade dos dados gerados pelo programa.

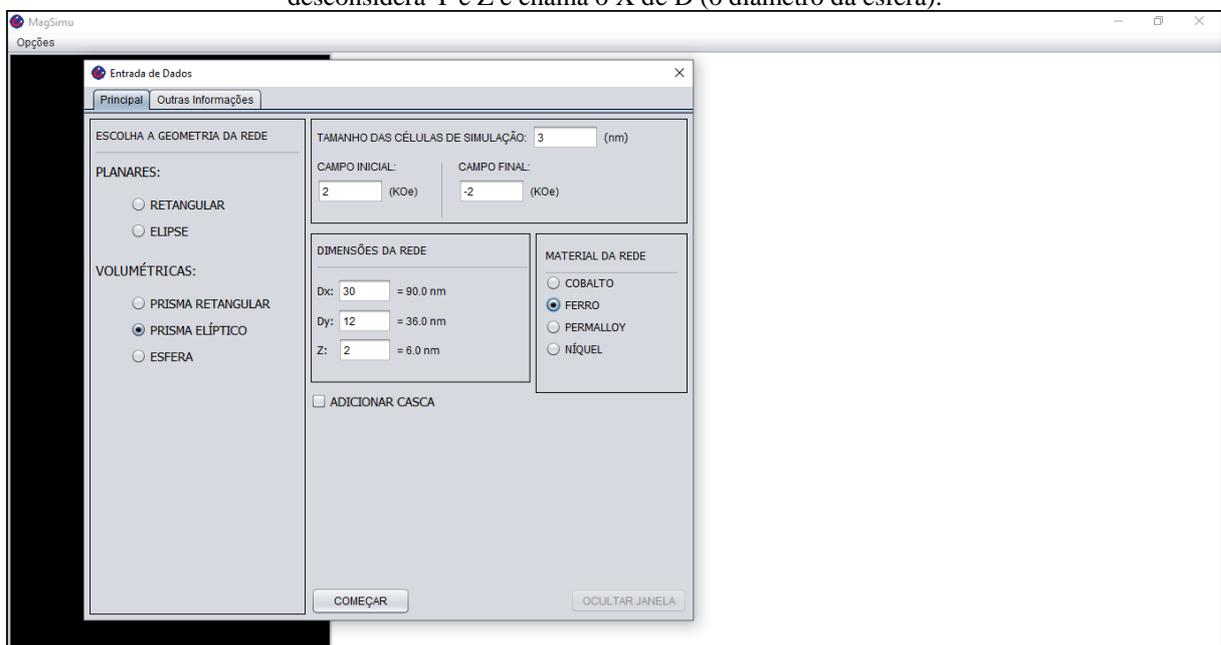
5. ESTRUTURA E FUNCIONAMENTO DO SOFTWARE MagSimu

5.1. APRESENTANDO O MagSimu

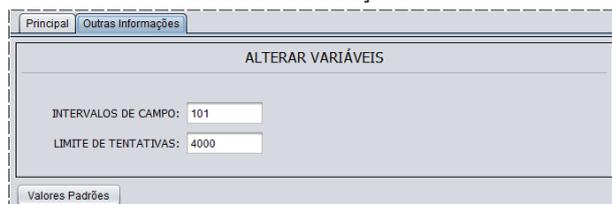
O MagSimu é um software de simulação, desenvolvido por nós, que calcula as condições de equilíbrio de um sistema magnético nanoestruturado sob o efeito de um campo externo variante, bem como pode obter sua magnetização, susceptibilidade, entre outras grandezas magnéticas para algumas geometrias específicas. Todas as condições iniciais e características da nanoestrutura são definidas pelo usuário no início do processo através da interface gráfica e, após isso, a simulação ocorrerá.

A interface gráfica do MagSimu é estruturada de acordo com a Figura 5.1, e a partir dela pode-se observar que, ao executar o programa, abrirá uma janela maximizada, com um espaço em branco para mostrar graficamente os estados do sistema magnético em processo, e outro espaço preto para apresentar os logs da operação; além de outra janela menor com um formulário no qual serão inseridas as informações relacionadas à simulação que virá em seguida.

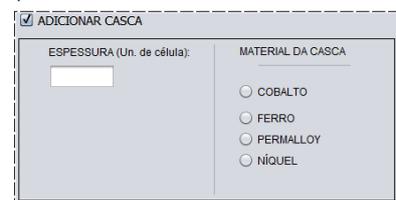
Figura 5.1 – Tela inicial do programa MagSimu. Os campos já estão preenchidos para iniciar um processo. A depender do tipo de geometria escolhida, a medida em Z será desconsiderada (se for planar), e as medidas em X e Y serão chamadas de Dx e Dy (diâmetros em X e Y) se forem escolhidas geometrias elípticas. A esfera desconsidera Y e Z e chama o X de D (o diâmetro da esfera).



Conteúdo da aba *Outras Informações*



Aparece se clicar em ADICIONAR CASCA



Fonte: Elaborado pelo autor.

Na janela “Entrada de Dados” na aba “Principal” o usuário dispõe das escolhas dos seguintes parâmetros:

a) Geometria da Rede

Esse parâmetro significa a escolha da forma geométrica do nanoelemento magnético. Para representar sistemas bidimensionais (planares), escolhe-se a forma retangular ou elíptica. Enquanto para sistemas tridimensionais (volumétricos), escolhe-se os prismas retangulares ou elípticos, ou esfera.

b) Tamanho das células de simulação

A escolha do tamanho das arestas das células cúbicas de simulação, em nm, deve levar em consideração as dimensões do nanoelemento, bem como o comprimento de troca do material em estudo. O tamanho da célula de simulação deve ser menor do que o comprimento de troca. Por exemplo, o comprimento de troca do Fe é da ordem de 10 nm, enquanto o do permalloy é 15 nm.

c) Campo inicial e campo final

Os valores do campo magnético no início e no final do processo de magnetização, em unidades de kOe, ao longo da componente x .

d) Dimensões da rede

Esse parâmetro consiste da escolha das dimensões físicas do nanoelemento magnético. A forma como são considerados varia de acordo com a geometria escolhida. O parâmetro Z só é disponibilizado para nanoestruturas volumétricas. Para o nanoelemento escolhido (planar, volumétrico, elíptico ou esférico), X (D_x para elípticos ou D para esféricos) é escolhido para ser o número de células ao longo do eixo x (ou o diâmetro da esfera), enquanto Y (D_y para elípticos) é escolhido para ser o número de células ao longo do eixo y . da mesma forma, Z é o número de células ao longo da altura (somente para os volumétricos).

e) Material da rede

O MagSimu reconhece os nanoelementos de Cobalto, Ferro, Permalloy ($Ni_{80}Fe_{20}$) e Níquel.

f) Adicionar casca

Esse parâmetro permite a simulação de sistemas constituídos de dois magnéticos, um no núcleo e outro na casca. Ao escolhê-lo, o usuário pode definir a espessura da casca e o material que é constituído.

Na aba “Outras Informações” o usuário pode definir em quantos intervalos prefere que o campo varie, indo do campo inicial até o campo final, além da quantidade máxima de vezes que o programa pode verificar se o torque é nulo ou suficientemente desprezível.

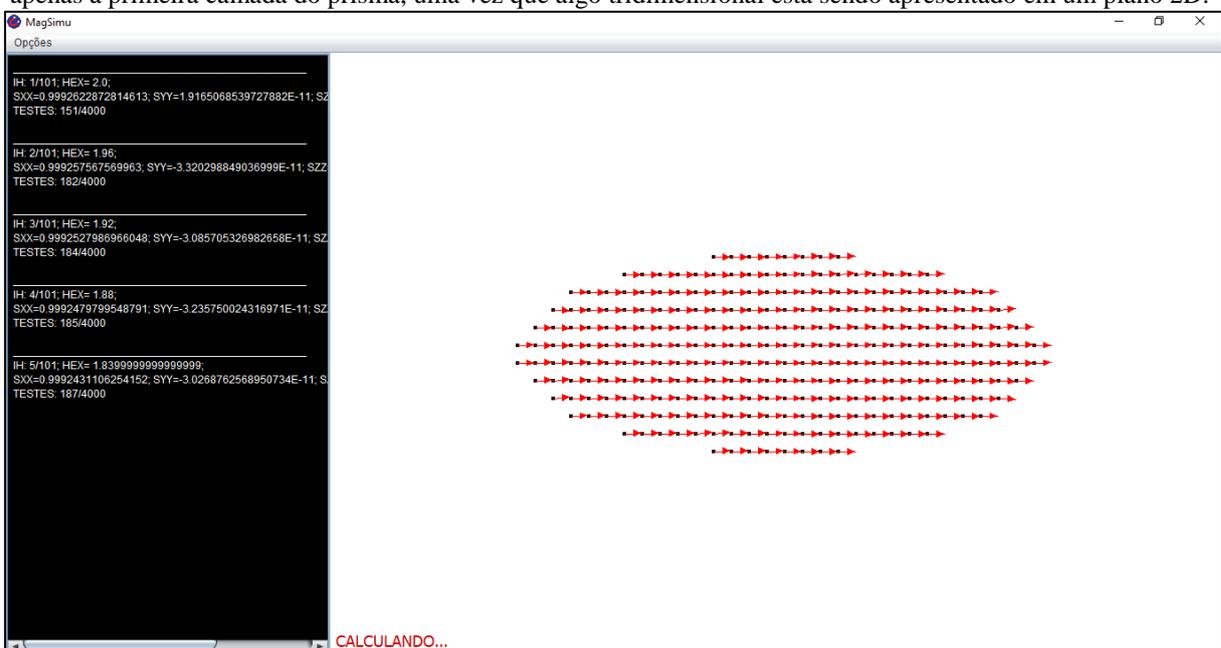
O menu “Opções” possui dois itens úteis que são o “Mostrar Entrada de Dados”, que exhibe a tela de entrada quando o usuário a oculta, e “Salvar Log”, que salva em arquivo “.txt” as informações mostradas na tela escura ao lado.

Ao inserir os dados e iniciar a simulação, o programa irá utilizar o método numérico auto-consistente para calcular as condições de equilíbrio do sistema para cada variação de campo externo, desde o valor inicial do campo até parar em um valor final (ambos escolhidos

pelo usuário). Observe na Figura 5.2 que serão calculadas as configurações de equilíbrio para um sistema magnético com formato de prisma elíptico de dimensões $30 \times 12 \times 2$ unidades de célula ($90 \times 36 \times 6 \text{ nm}$, dado que a célula de simulação neste exemplo tem arestas iguais a 3 nm), observando que 30 e 12 são os valores dos diâmetros maior e menor da base elíptica do prisma, e é constituído inteiramente de permalloy ($\text{Ni}_{80}\text{Fe}_{20}$). O campo começará em 2 kOe e terminará em -2 kOe variando em intervalos escolhidos na aba “Outras informações” (o número de intervalos padrão do programa é 101, mas o usuário pode mudar esse valor). Assim que a primeira configuração de equilíbrio for calculada, será desenhada na tela branca uma matriz de tamanho $X \times Y$, sendo os pontos as células de simulação da primeira camada do prisma visto de cima, e sobre os pontos, os vetores momento magnético de cada uma apontando para uma direção em que sua energia é mínima.

Ao marcar a opção “Adicionar casca” na tela de entrada, o espaço vazio logo abaixo dessa opção será preenchido por um painel com um campo de texto para informar a espessura da casca e um grupo de opções para escolher o material da casca (ferro, cobalto, permalloy ou níquel). Todo o funcionamento do programa será melhor explicado nas seções seguintes.

Figura 5.2 – Programa operando com as configurações da Figura 5.1. O mapa de vetores na figura refere-se apenas à primeira camada do prisma, uma vez que algo tridimensional está sendo apresentado em um plano 2D.



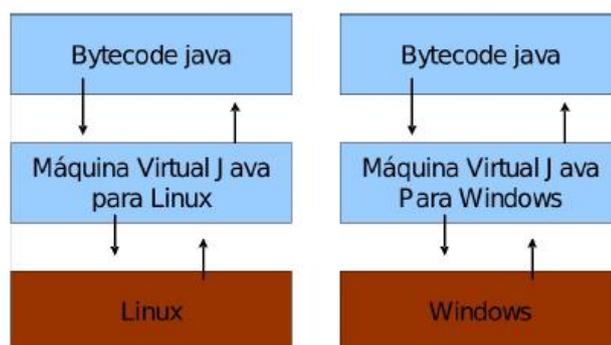
Fonte: Elaborado pelo autor.

5.2. TECNOLOGIA E MÉTODOS UTILIZADOS

5.2.1. Java

O Java foi criado pela antiga Sun Microsystems e mantida através de um comitê (<http://www.jcp.org>) (CAELUM, 2018). O Java é uma linguagem de programação orientada a objetos, além de ser compilável, mas diferente das outras linguagens que têm seu código fonte compilado para o código de máquina específico de uma plataforma e sistema operacional, o Java é compilado por uma máquina virtual (JVM) já presente no computador (Figura 5.3) e é capaz de executar em outras plataformas sem problemas. Esta linguagem é indicada para aplicações de médio a grande porte e também possui uma estrutura bastante organizada, o que torna extremamente mais rápido e fácil de fazer alterações, desde que se siga as boas práticas e recomendações sobre design orientado a objeto.

Figura 5.3 – Compilação de programas em Java pela JVM.



Fonte: CAELUM, 2018.

Por ser orientado a objeto, ele traz a ideia da representação de cada elemento em termos de um objeto ou classe. Esse tipo de representação procura aproximar o sistema que está sendo criado ao que é observado no mundo real. Esse tipo de representação traz algumas vantagens como a reutilização de código, o que diminui o tempo de desenvolvimento e a facilidade de leitura e manutenção do código.

A linguagem Java foi a escolhida para o desenvolvimento do programa, uma vez que este software tem uma complexidade considerada de médio porte, com diversos componentes que tenderão a aumentar de quantidade com a evolução do programa, o que tornou seu uso mais viável e organizado para o desenvolvimento.

Outro ponto forte na escolha do Java tem relação com a quantidade enorme de bibliotecas gratuitas para realizar os mais diversos trabalhos. Graças a isso e se aliando às várias

ferramentas que o NetBeans IDE 8.2 (será explicado mais a frente) oferece, foi possível construir a interface gráfica e vários outros recursos muito mais facilmente e sem muito esforço. Claro que muitas linguagens têm os recursos necessários para a construção deste programa, mas nem todos os recursos são gratuitos.

Os termos comuns em Java que serão usados neste trabalho são:

- **Classe:** classe se refere a um arquivo que possui um agrupamento de informações, ou seja, ele possui um conjunto de atributos (variáveis) e métodos (funções) em seu código que servem para uma finalidade específica.
- **Instância:** é, basicamente, a criação de um objeto a partir de uma classe.
- **Objeto:** um objeto é algo distinguível que contém os atributos e os métodos da classe, mas que, através dele, poderão ser usados em algum processo. Cada objeto tem uma identidade e é distinguível de outro mesmo que seus atributos sejam idênticos, isso quer dizer que podem ser criados vários objetos diferentes de uma mesma classe.
- **Herança:** através da herança, as subclasses (aquelas que herdam) podem usar os atributos e métodos que uma classe-mãe possui. Desse modo, se houver muitas classes que utilizam quase os mesmo atributos e métodos, é só fazê-las herdar de uma única classe o que há em comum entre elas.

5.2.2. NetBeans IDE 8.2

Para construir o programa, foi escolhida a IDE NetBeans por ser uma ferramenta que auxilia em várias tarefas, como escrever, compilar, depurar e instalar aplicações. O NetBeans foi criado com a intenção de tornar o desenvolvimento mais simplificado para o programador e aumentar a produtividade deste. E além da linguagem Java, oferece suporte também para várias outras linguagens de programação.

O NetBeans comporta um grande conjunto de recursos em si, como bibliotecas, módulos e API's (conjuntos de rotinas, protocolos e ferramentas para a construção de aplicativos), sendo que disponibiliza uma vasta documentação de forma bem organizada. Com todos estes artifícios, o desenvolvimento de um software se torna bem mais rápido e prático. Atualmente está disponível em diversos idiomas, tornando-o cada vez mais popular e facilitando o acesso, tanto para iniciantes em programação, quanto para experientes da área.

Esta IDE disponibiliza vários recursos de organização para o código, possibilitando monitorar o funcionamento do código em tempo de execução, verificar onde o código mostra

conflitos e erros enquanto operamos o programa, e assim fazer a correção desses erros de execução.

Para o desenvolvimento da interface gráfica do software, o NetBeans possui o recurso de clicar e arrastar os componentes visuais (campos de texto, botões, painéis, entre outros) para a tela, o que auxiliou na criação das telas que o sistema possui, facilitando o tratamento das informações que serão apresentadas para o usuário e aumentando a produtividade na construção do programa.

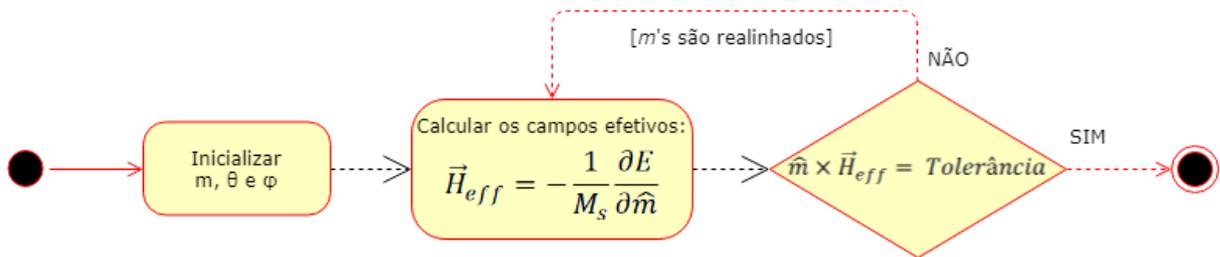
5.2.3. Método Numérico Auto-Consistente

Este é um método implementado por meio de recursos computacionais. A partir do conceito de célula de simulação, explicado anteriormente, ele calcula a configuração de equilíbrio de um sistema magnético nanoestruturado alinhando os momentos magnéticos das células com o campo efetivo. Para isso, este algoritmo realiza os seguintes passos para qualquer sistema de dimensões nanométricas:

1. Dado um sistema magnético, inicializaremos todas as variáveis $(m_1^{x0}, m_1^{y0}, m_1^{z0}, m_2^{x0}, m_2^{y0}, m_2^{z0}, \dots, m_n^{x0}, m_n^{y0}, m_n^{z0})$ do sistema em uma configuração magnética de saturação na direção de um campo magnético externo aplicado;
2. A partir das variáveis $m^{0's}$, calculamos o campo médio local (\vec{H}_{eff}) sobre cada uma das células;
3. Tendo o campo médio, calculamos o torque magnético sobre cada uma das células ($\vec{\tau} = \vec{m}_0 \times \vec{H}_{eff}$);
4. Checamos se o torque é suficientemente desprezível, ou seja, menor que 10^{-6} , em cada uma das células;
5. Se a condição acima não for satisfeita. Colocamos todas as variáveis na direção do campo médio local e obtemos um novo conjunto de variáveis $m's$ $(m_1^x, m_1^y, m_1^z, m_2^x, m_2^y, m_2^z, \dots, m_n^x, m_n^y, m_n^z)$. Nesse caso, voltamos ao 1º passo, guardamos as variáveis $m's$ nas variáveis $m^{0's}$ e refazemos o cálculo, iterativamente, até que a condição do passo 5, seja satisfeita;
6. Quando a condição for satisfeita, o cálculo numérico encontrou a configuração de equilíbrio, que é o estado de mais baixa energia magnética.

O fluxograma da Figura 5.4 descreve de forma mais ilustrativa os passos do algoritmo.

Figura 5.4 – Fluxograma do método numérico auto-consistente.



Fonte: Elaborado pelo autor.

5.3. REQUISITOS DO MAGSIMU

O MagSimu deve atender aos seguintes requisitos:

1. Possuir uma interface gráfica simples, intuitiva e fácil de usar, uma vez que se destina a usuários que não sabem programar e com o conhecimento suficiente para fazer este tipo de simulações.
2. Calcular, a partir do método auto-consistente, as configurações de equilíbrio de um sistema magnético, com um campo magnético externo variante.
3. Salvar, em arquivo com um formato legível para o OriginLab 9.0, os dados gerados sobre os momentos magnéticos das células de simulação em cada configuração de equilíbrio, que foram calculados em cada intervalo de campo magnético.
4. Calcular a magnetização do sistema em cada intervalo e salvá-las em arquivo com um formato legível para o OriginLab 9.0.
5. Possuir código aberto para possibilitar que modificações no programa possam ser feitas de acordo com as necessidades de pesquisa.

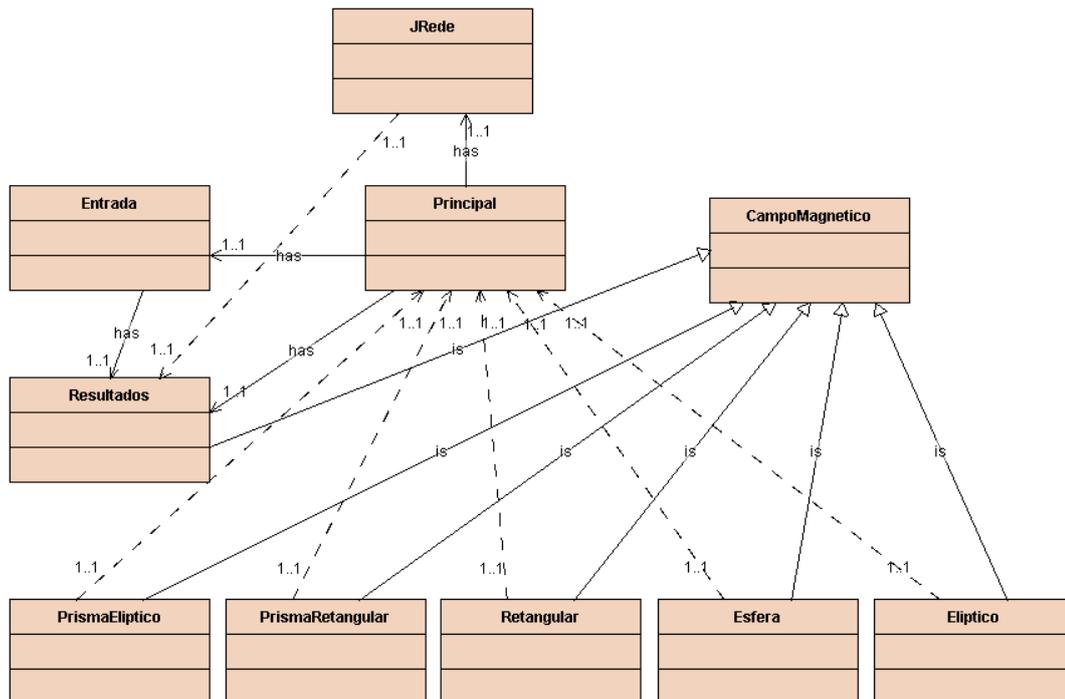
5.4. ANÁLISE DOS DIAGRAMAS

Nesta seção, serão apresentadas as estruturas que compõem o MagSimu e analisadas a função de cada uma delas no processo. Para esta análise, serão utilizados o diagrama de classes e o diagrama de atividade. Mais informações sobre cada tipo serão mostradas no decorrer do texto.

5.4.1. Diagrama de classes

O diagrama de classes mostra a forma como estão representadas a estrutura e as relações das classes que servem de modelo para os objetos de um programa. Através dele, a forma como cada classe funciona em relação aos demais poderá ser esclarecida.

Figura 5.5 – Diagrama de classes simplificado do software MagSimu.



Fonte: Elaborado pelo autor.

O diagrama acima mostra como o programa MagSimu está organizado por meio da estrutura de suas classes e as relações entre elas. É comum, na maioria dos diagramas, que sejam mostrados os atributos e métodos nesses espaços abaixo dos nomes das classes, mas como ficariam muito grandes e desorganizados se fossem exibidos, resolvi mostrar os detalhes de cada classe no apêndice A. A seguir, este diagrama será analisado classe por classe, enquanto suas funções e relações serão mostradas.

- **CampoMagnetico**: Esta é uma classe que só tem o objetivo de passar os seus atributos, por herança, para as seguintes classes: **Resultados**, **Retangular**, **Eliptico**, **PrismaRetangular**, **PrismaElptico** e **Esfera**. As setas que interligam a classe do **CampoMagnético** as demais classes definem essa relação de herança entre elas. Se não fosse pela herança, seria necessário declarar os mesmos atributos em todas essas classes, o que seria uma tarefa cansativa e complicada as vezes.

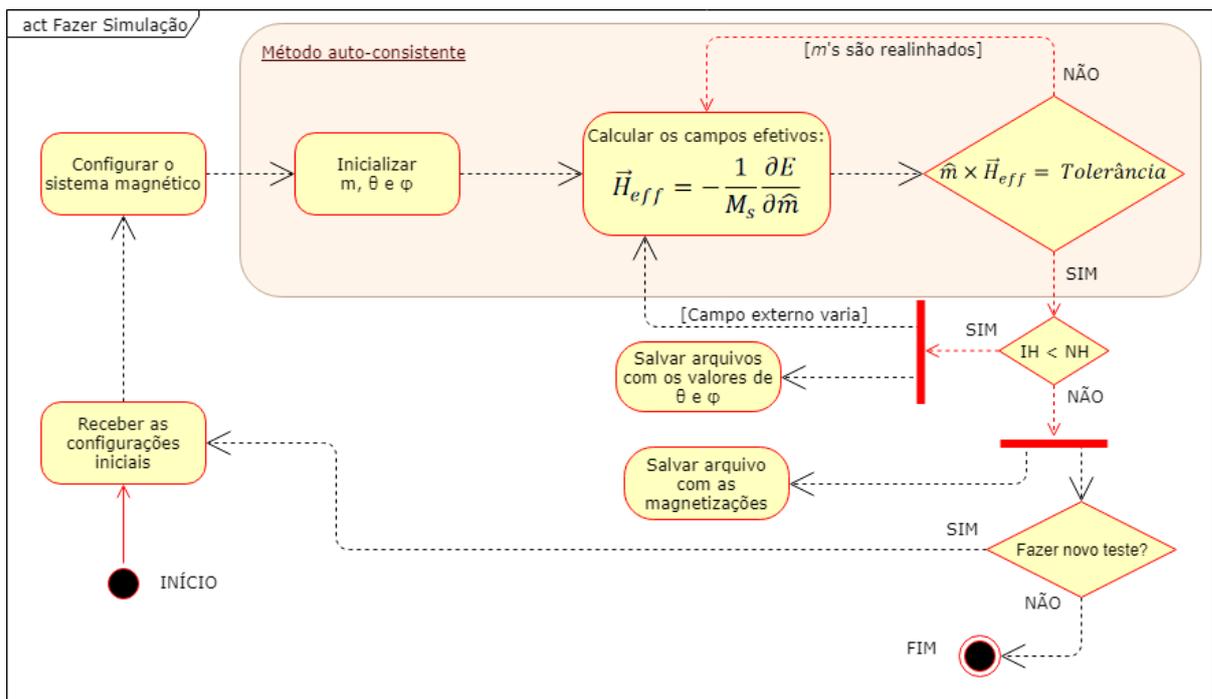
- **Resultados**: A função desta classe é basicamente a de armazenar todas as informações necessárias geradas no processo, tanto para serem usadas entre as classes quanto para serem gravadas em arquivo. Apenas um objeto de **Resultados** é instanciado em **Entrada**, esse objeto é então utilizado pelas demais classes através de **Principal**, até o fim do processo. Esta classe está relacionada a todas as outras, uma vez que todas dependem das informações que ela possui.
- **Entrada**: esta classe é a responsável pela tela de entrada de dados, na qual o usuário define as características do sistema e as condições iniciais. Sua relação com **Resultados** ocorre porque todos os dados do usuário são mandados para o seu objeto, que é instanciado em **Entrada**, e logo em seguida, junto com todas as informações do usuário, enviado para **Principal**, onde será usada em todos os processos menores até o fim. Sua relação com **Principal** se dá pelo fato de **Entrada** ser instanciada e aberta por meio de **Principal**. Esta classe também é responsável por dar início a simulação, uma vez que ela realiza a chamada dessa função, que está em **Principal**, e dá a vez a este.
- **Principal**: Ela será a responsável pela maioria dos processos que ocorrem. A partir das informações do objeto do tipo **Resultados**, quando **Entrada** chamar a função responsável por iniciar a simulação, **Principal** irá instanciar um objeto de uma das seguintes classes: **Retangular**, **Elíptico**, **PrismaRetangular**, **PrismaElíptico** e **Esfera**, a depender da geometria escolhida pelo usuário para o sistema magnético. Toda vez que o objeto de uma dessas classes executar o método para encontrar a configuração de equilíbrio, **Principal** irá fazer o objeto executar o mesmo cálculo novamente para um novo valor de campo até chegar a um valor final escolhido. Junto desses processos repetitivos, a classe também grava arquivos com os valores de θ e φ para cada configuração de equilíbrio para análises futuras, além de um arquivo com a magnetização do sistema ao final de todo o processo. Vale lembrar que essa é a classe principal do programa e a responsável por iniciar e fechar o programa por meio da interface gráfica gerada por ela.
- **JRedes**: Esta é uma classe responsável apenas por desenhar os pontos e setas que aparecem na interface gráfica gerada por **Principal**. Ela acessa os valores das dimensões e dos momentos magnéticos em **Resultados** e os usa para representá-los na tela a cada variação do campo.

- ****Demais Classes***: são estas classes que possuem o método numérico auto-consistente e calculam a configuração de equilíbrio dos sistemas escolhidos. Como cada uma tem formatos geométricos diferentes, o método numérico tem adaptações específicas para elas, o que possibilita o cálculo adequado desses sistemas. Outro ponto é que elas são as responsáveis por escrever os logs naquela tela preta lateral em cada variação de campo.

5.4.2. Diagrama de atividade

Este tipo de diagrama é voltado para modelar métodos, algoritmos ou até mesmo um processo completo. Ele representa os fluxos conduzidos por processamentos, ou melhor, são os métodos correspondentes às operações sobre as classes. É essencialmente um gráfico de fluxo, mostrando o fluxo de controle de uma atividade para outra.

Figura 5.6 – Diagrama de atividade do software MagSimu.



Fonte: Elaborado pelo autor.

O diagrama da Figura 5.6 mostra o funcionamento do programa MagSimu por meio dos processos que acontecem do início ao fim. A seguir, os processos serão analisados e esclarecidos de acordo com o diagrama.

- ***Receber as configurações iniciais***: Esta etapa se trata da entrada das informações que o usuário deseja usar para fazer a simulação. O usuário deverá informar todas as

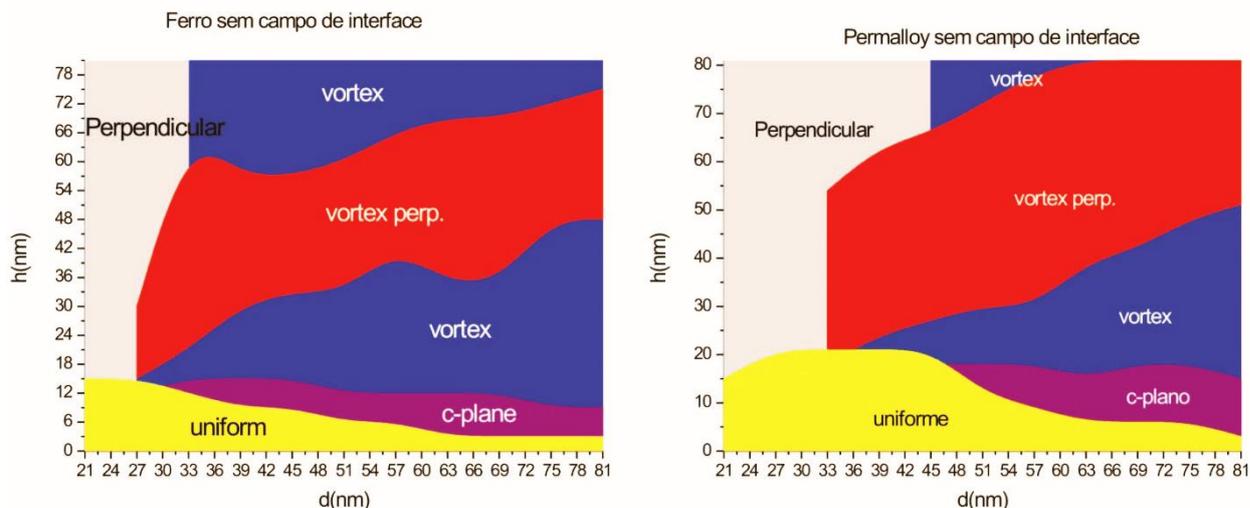
características que a nanoestrutura assumirá, como geometria, dimensões, composição, se irá ter casca e, se tiver, qual a sua espessura e composição; assim como as condições do meio, como intensidades inicial e final do campo externo, tamanho da célula de simulação e o número de intervalos que o campo irá variar.

- **Configurar o sistema magnético:** Após receber todas as informações do usuário, o sistema magnético será configurado com base nessas informações para dar início aos cálculos.
- **Método numérico auto-consistente:** Todos os processos dentro deste método são explicados na seção 5.2.3.
- **Nó de decisão $IH < NH$:** O número de intervalos que o campo vai variar é NH , já o IH é a quantidade de intervalos de campo que foram processados. Enquanto o número de intervalos não forem completados, ou seja, igual a NH , o programa continuará calculando as configurações de equilíbrio para cada variação do campo magnético.
- **Salvar arquivos com os valores de θ e φ :** Quando cada configuração de equilíbrio for calculada, é necessário guardar as informações das direções dos momentos magnéticos do sistema para aquele caso para que possam ser analisados pelo usuário. Os dados são gravados em arquivos “.dat” de modo que sejam legíveis para o programa OriginLab 9.0.
- **Salvar arquivos com as magnetizações:** Após o campo chegar ao seu valor final, os dados de magnetização (a média de todos os momentos magnéticos do sistema) de cada intervalo serão salvos em um arquivo “.dat” de modo que também sejam legíveis para o OriginLab 9.0.
- **Fazer novo teste:** Após todo o processo acabar, é possível fazer uma nova simulação com novas informações. Caso não seja desejado um novo teste, o usuário pode fechar o programa.

6. AVALIAÇÃO DOS RESULTADOS

Para verificar a confiabilidade dos resultados gerados pelo MagSimu, é necessário fazer comparativos com dados existentes de autorias de maior confiança, de modo que a convergência entre eles garanta a confirmação de que os resultados obtidos no programa sejam corretos. No nosso caso, utilizaremos o diagrama de fase presente no trabalho de SILVA (2011), que pode ser visto na Figura 6.1.

Figura 6.1 – Diagrama de fase de discos (prismas cilíndricos) de ferro e de permalloy baseado na influência do diâmetro e da espessura. Note que, dependendo das medidas destes últimos, a direção dos momentos magnéticos pode se distribuir no sistema de modo a apresentar diferentes fases.

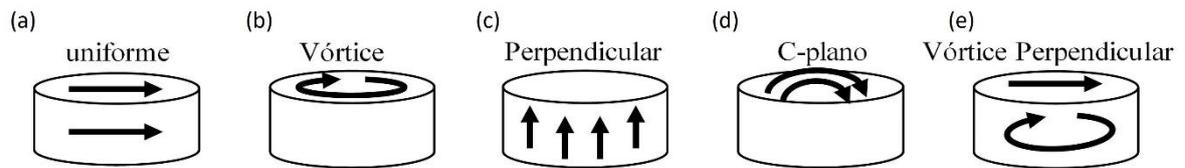


Fonte: SILVA, 2011.

Na Figura 6.1, são apresentadas as informações sobre as fases do Ferro (a esquerda) e do Permalloy (a direita). As regiões coloridas nos diagramas mostram que tipo de fase o disco terá quando forem determinados o seu diâmetro e espessura. Dependendo das medidas do disco, após a estrutura passar pelo processo em que um campo magnético externo magnetiza os momentos magnéticos intrínsecos e os satura, e depois, a intensidade do campo é diminuída até zero, observa-se que os momentos magnéticos se distribuirão de modo a formar uma fase, que pode ser:

- a) **Uniforme (*uniform*)**: a fase uniforme é aquela em que todos os momentos magnéticos estão alinhados e apontando para a mesma direção (ver Figura 6.2.a).
- b) **Vórtice (*vortex*)**: os momentos magnéticos se distribuem de modo a tomar uma forma circular, ou seja, um vórtice, que tem seu eixo de rotação perpendicular e central à face do disco (ver Figura 6.2.b).
- c) **Perpendicular (*Perpendicular*)**: quando a medida da espessura é muito maior que a medida do diâmetro, os momentos magnéticos tendem a se alinhar uniformemente na direção em que o disco está alongado (ver Figura 6.2.c).
- d) **C-plano (*c-plane*)**: esta é uma fase entre as fases uniforme e vórtice, e os momentos magnéticos se arranjam de modo a formar um C, mas ainda permanecem alinhadas com um mesmo plano (ver Figura 6.2.d).
- e) **Vórtice Perpendicular (*vortex perp.*)**: semelhante à fase vórtice explicado anteriormente, mas com a variação de que, nesse caso, o eixo de rotação do vórtice é paralelo à face do disco (ver Figura 6.2.e).

Figura 6.2 – Ilustrando as fases magnéticas de um disco.



Fonte: Elaborado pelo autor.

Para verificarmos se as simulações feitas no MagSimu geram resultados confiáveis e condizentes com as informações da Figura 6.1, fizemos quatro experimentos, sendo dois deles usando o Ferro, e os outros dois usando o Permalloy. Em cada experimento, foram feitas simulações para discos utilizando medidas diferentes de diâmetro e espessura, mas com as mesmas condições (tamanho da célula de simulação, número de intervalos, campo inicial e final). Com os resultados gerados, faremos um estudo de fase e verificamos se estão de acordo com os diagramas da Figura 6.1. É importante dizer que as nanoestruturas as quais são trabalhadas nas experiências (discos) são o mesmo que prismas elípticos de diâmetros iguais, e por esse motivo, foi escolhido esse tipo de geometria para os experimentos, mas com a condição de que os diâmetros terão o mesmo valor.

Como dito anteriormente, as configurações que usamos no programa em cada experimento são as mesmas, exceto pelo material e pelas medidas escolhidas. As informações escolhidas na tela de entrada de dados do programa estão apresentadas na Figura 6.3. As simulações foram feitas em um notebook, que possui um poder de processamento relativamente baixo se comparado com outros tipos. O computador usado para os testes, tem um sistema operacional Windows 10 de 64 bits, processador Intel Core i3 e memória RAM de 4GB.

Figura 6.3 – Tela com as informações que os experimentos terão em comum. Observe que, como é um disco, a geometria usada foi o prisma elíptico com o detalhe que os diâmetros são iguais, ou seja, ele é circular.

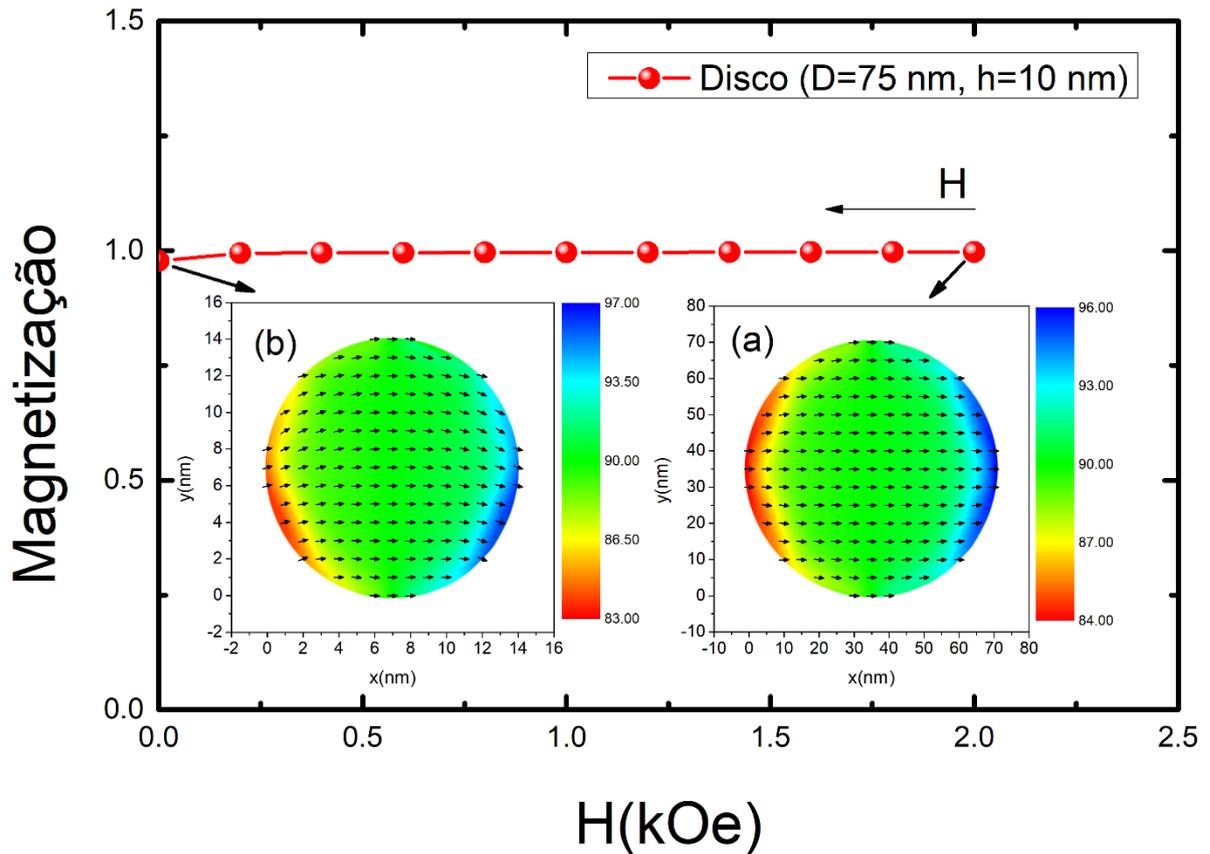
Fonte: Elaborado pelo autor.

6.1. EXPERIÊNCIA 1:

Na primeira experiência, escolhemos um disco de Ferro com diâmetro de 75 nm e espessura de 10 nm (15x2 em unidades de célula cúbica de simulação). As outras informações da simulação estão exibidas na Figura 6.3. Note que, de acordo com a Figura 6.1, o sistema magnético escolhido deve apresentar uma fase C-Plano quando chegar a 0 kOe.

Com relação ao tempo gasto para concluir a simulação, sabendo que o programa está considerando 322 células ao todo no sistema magnético escolhido, o processo todo demorou cerca de 2 minutos, 53 segundos e 522 milissegundos, o que levaria ainda menos em computadores com poder de processamento elevado.

Figura 6.4 – Gráfico de magnetização do disco de Ferro (75nm × 10nm) com os mapas de vetores da primeira camada de células do disco para $H = 2$ kOe e $H = 0$ kOe.



Fonte: Elaborado pelo autor.

Como esperado, a magnetização está saturada em um campo de 2 kOe, o que o faz apresentar uma fase uniforme (ver Figura 6.4.a), mas quando o campo aplicado chega a 0 kOe, o sistema apresenta uma pequena curvatura no alinhamento dos momentos magnéticos (ver Figura 6.4.b), que é um arranjo típico da fase C-Plano.

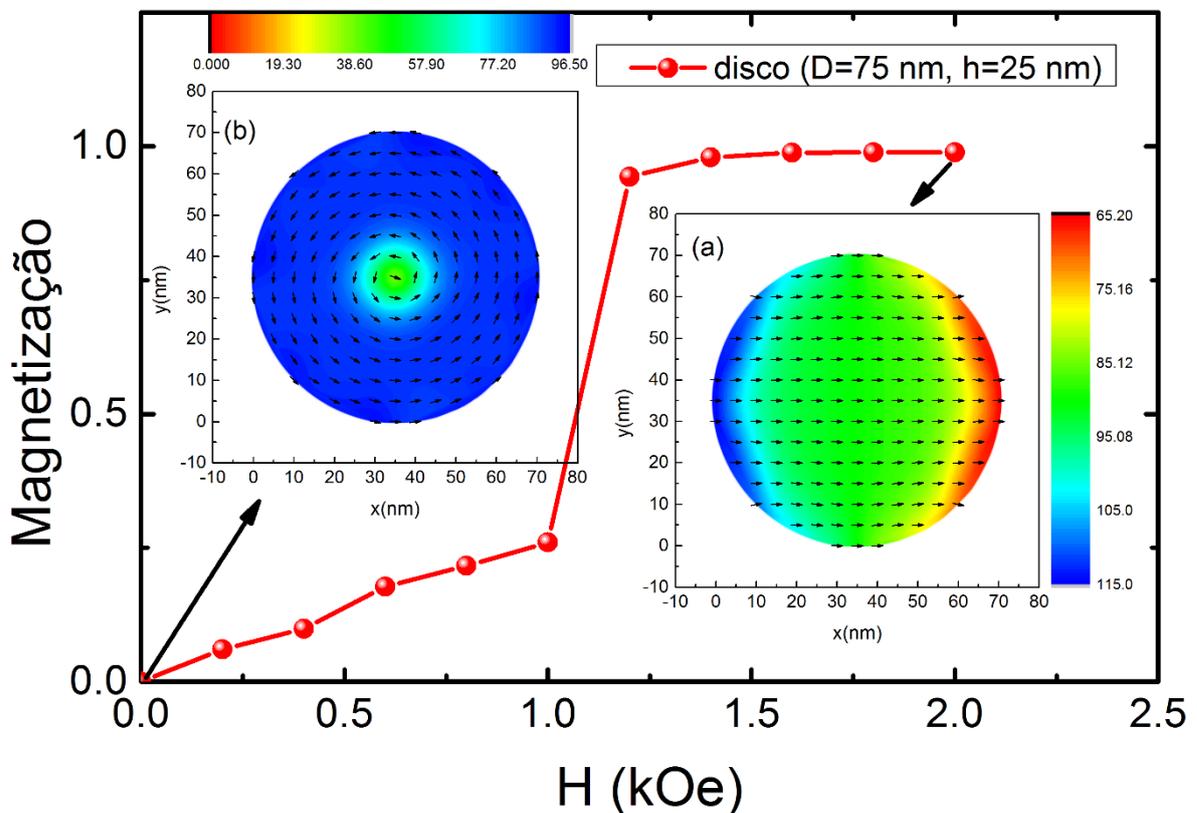
Desse modo, confirmamos que o sistema escolhido nesta experiência conseguiu formar uma fase C-Plano e conseqüentemente, tanto o cálculo no programa quanto os resultados obtidos a partir dele, pelo menos para este caso, são confiáveis e seguros para utilização e análise.

6.2. EXPERIÊNCIA 2:

Na segunda experiência, escolhemos outro disco de Ferro com diâmetro de 75 nm e espessura de 25 nm (15x5 em unidades de célula cúbica de simulação). As outras informações da simulação são as mesmas da Figura 6.3. Agora dessa vez, de acordo com a Figura 6.1, o sistema magnético escolhido deve apresentar uma fase de vórtice quando chegar a 0 kOe.

Com relação ao tempo gasto para concluir a simulação, com o aumento na medida da espessura em 3 unidades de célula (15 nm) com relação ao sistema anterior, estamos considerando 483 células a mais no cálculo (805 células ao todo), o que fez o processo todo demorar cerca de 30 minutos, 55 segundos e 132 milissegundos. Esse aumento significativo no tempo gasto ocorreu, principalmente, devido ao cálculo do campo dipolar, que relaciona cada célula com todas as outras, juntamente com as tentativas de tornar o torque nulo ou pequeno o suficiente para ser desprezível, pois repete os cálculos a cada fracasso.

Figura 6.5 – Gráfico de magnetização do disco de Ferro (75nm × 25nm) com os mapas de vetores da primeira camada de células do disco para (a) $H = 2$ kOe e (b) $H = 0$ kOe.



Fonte: Elaborado pelo autor.

Como também era esperado, a magnetização está saturada em um campo de 2 kOe, o que o faz apresentar uma fase uniforme (ver Figura 6.5.a), mas quando o campo aplicado chega a 0 kOe, os momentos magnéticos do sistema se distribuem de modo a formar um vórtice (ver Figura 6.5.b).

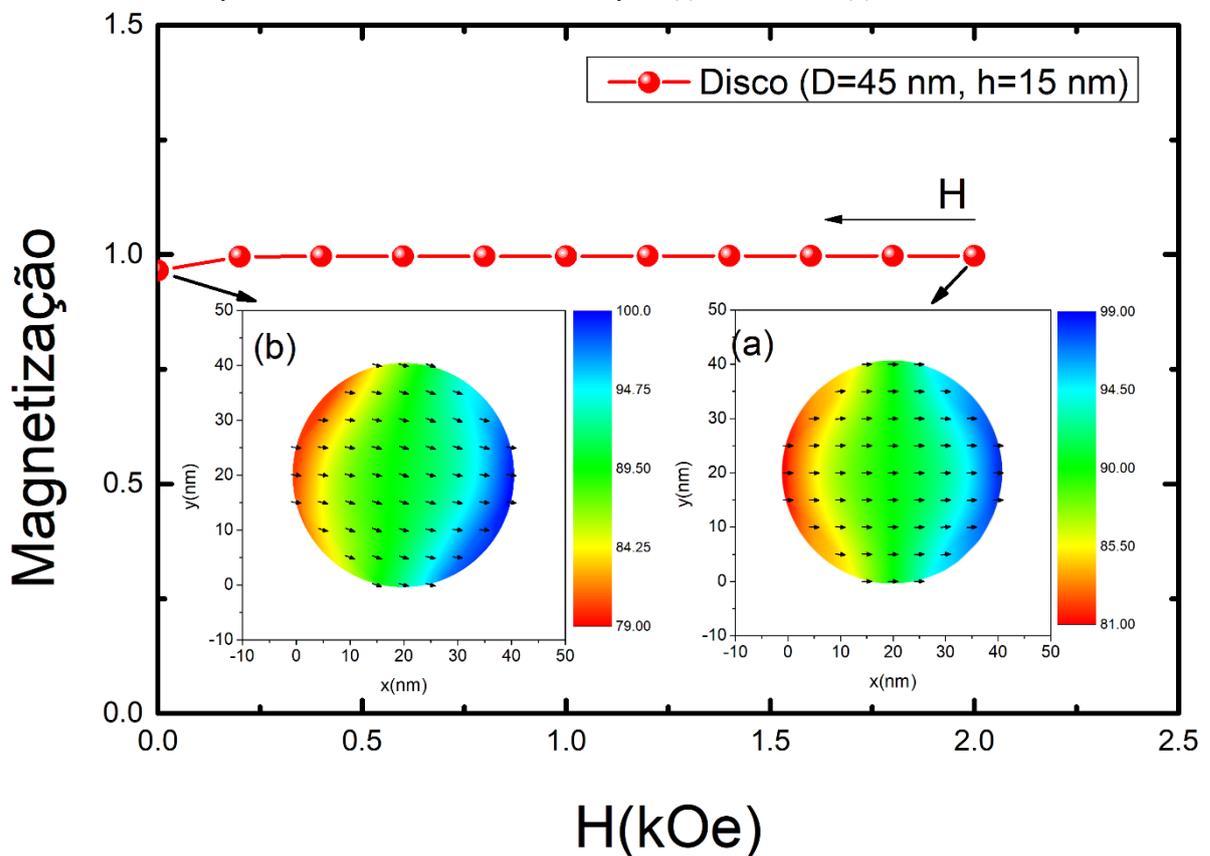
Com essa segunda experiência, confirmamos que o sistema escolhido nesta experiência conseguiu formar uma fase de vórtice e conseqüentemente, tanto o cálculo no programa quanto os resultados obtidos a partir dele, ao menos para esses dois casos, são confiáveis e seguros para utilização e análise.

6.3. EXPERIÊNCIA 3:

Na terceira experiência, escolhemos um disco de Permalloy com diâmetro de 45 nm e espessura de 15 nm (9x3 em unidades de célula cúbica de simulação). As outras informações da simulação estão exibidas na Figura 6.3. Note que, de acordo com a Figura 6.1, o sistema magnético escolhido deve apresentar uma fase uniforme quando chegar a 0 kOe.

Com relação ao tempo gasto para concluir a simulação, sabendo que o programa está considerando 171 células ao todo no sistema magnético escolhido, o processo todo demorou cerca de 56 segundos e 875 milissegundos.

Figura 6.6 – Gráfico de magnetização do disco de Permalloy (45nm × 15nm) com os mapas de vetores da primeira camada de células do disco para (a) $H = 2$ kOe e (b) $H = 0$ kOe.



Fonte: Elaborado pelo autor.

Assim como as experiências anteriores, esta também está correta, pois quando a magnetização está saturada em um campo de 2 kOe, o sistema apresenta uma fase uniforme (ver Figura 6.6.a), e quando o campo aplicado chega a 0 kOe, os momentos magnéticos do sistema mantem seu alinhamento (continua com uma fase uniforme), mesmo que sua direção tenha mudado um pouco (ver Figura 6.6.b). Para explicar essa mudança na direção dos momentos magnéticos, temos que o Permalloy não possui anisotropia, o que significa que ele

não tem uma direção privilegiada que o mantenha apontando para ela, por este motivo que os momentos magnéticos não permaneceram apontando na direção do campo, mas continuaram alinhados uniformemente entre si.

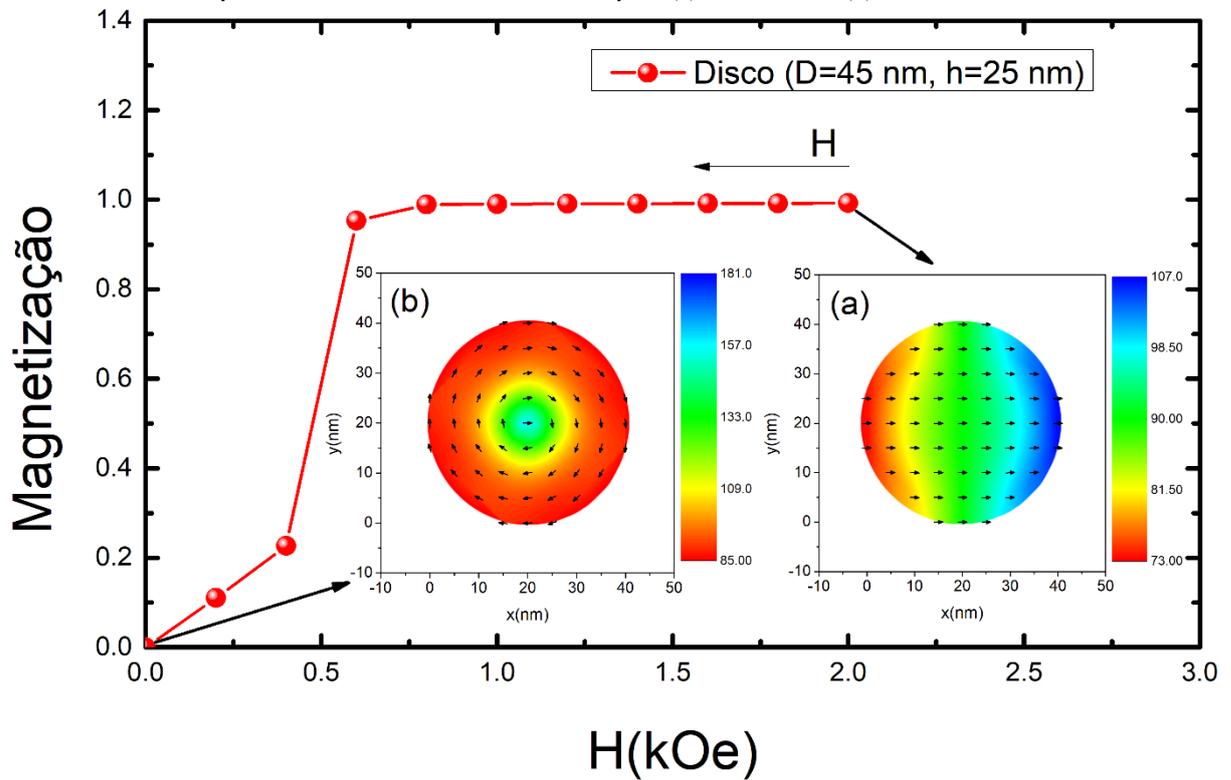
Com essa terceira experiência, confirmamos que o sistema escolhido nesta experiência conseguiu manter uma fase uniforme e conseqüentemente, tanto o cálculo no programa quanto os resultados obtidos a partir dele, ao menos para esses dois casos, são confiáveis e seguros para utilização e análise.

6.4. EXPERIÊNCIA 4:

Na quarta e última experiência, escolhemos um disco de Permalloy com diâmetro de 45 nm e espessura de 25 nm (9x5 em unidades de célula cúbica de simulação). As outras informações da simulação são as mesmas da Figura 6.3. Dessa vez, de acordo com a Figura 6.1, o sistema magnético escolhido deve apresentar uma fase de vórtice quando chegar a 0 kOe.

Com relação ao tempo gasto para concluir a simulação, com o aumento na medida da espessura em 2 unidades de célula (10 nm) com relação ao sistema anterior, estamos considerando 114 células a mais no cálculo (285 células ao todo), o que fez o processo todo demorar cerca de 2 minutos, 21 segundos e 842 milissegundos. Esse aumento no tempo gasto, como explicado no experimento 2, ocorreu, principalmente, devido ao cálculo do campo dipolar e as tentativas de tornar o torque nulo ou pequeno o suficiente para ser desprezível.

Figura 6.7 – Gráfico de magnetização do disco de Permalloy ($45\text{nm} \times 25\text{nm}$) com os mapas de vetores da primeira camada de células do disco para (a) $H = 2\text{ kOe}$ e (b) $H = 0\text{ kOe}$.



Fonte: Elaborado pelo autor.

Como também era esperado, a magnetização está saturada em um campo de 2 kOe, o que o faz apresentar uma fase uniforme (ver Figura 6.7.a), mas quando o campo aplicado chega a 0 kOe, os momentos magnéticos do sistema se distribuem de modo a formar um vórtice (ver Figura 6.7.b).

Com essa última experiência, confirmamos que, além do sistema escolhido nesta experiência ter conseguido formar uma fase de vórtice, os cálculos feitos no programa geram resultados corretos, confiáveis e seguros para utilização e análise.

7. CONCLUSÃO

Este trabalho apresentou o software MagSimu, uma ferramenta de simulação de sistemas magnéticos nanoestruturados. Por meio deste estudo, mostrou-se quais são os objetivos iniciais para o desenvolvimento deste programa e quais os requisitos que o software deveria atender.

Uma vez definidos os objetivos, foram levantados os seguintes requisitos: possuir uma interface gráfica simples e de fácil uso; calcular as configurações de equilíbrio de um sistema magnético, pelo método auto-consistente, com um campo externo variante; salvar os resultados do programa em arquivos que podem ser analisados no OriginLab 9.0 (ou em qualquer outro que tenha a mesma finalidade); e possuir código aberto. Após a conclusão do software, todos os requisitos foram atendidos e os objetivos alcançados, o que significa que, com um conhecimento suficiente em nanomagnetismo, qualquer usuário é capaz de manipular esta ferramenta com facilidade.

Este programa foi construído ao utilizar uma linguagem de programação orientada a objetos, o Java, para a implementação da interface gráfica e das funcionalidades. O desenvolvimento do seu código-fonte em Java foi feito do início ao fim no NetBeans 8.2, desde os cálculos à parte visual. Os resultados obtidos com o programa são gravados em arquivos “.dat” em um formato legível para o OriginLab 9.0, que é responsável por gerar curvas de histerese, mapas das fases magnéticas, entre outras possibilidades.

Os experimentos feitos tiveram a finalidade de comprovar a confiabilidade dos resultados que o programa gera em suas simulações. Uma vez que as fases magnéticas das nanoestruturas estão de acordo com as informações de SILVA (2011), foi possível declará-las como corretas, o que significa que não há problemas nos cálculos que o programa realiza.

Alguns aspectos negativos observados neste programa estão relacionados com a baixa diversidade de recursos para serem usados em uma simulação, como os tipos de materiais e geometrias, bem como o tempo gasto para simular sistemas magnéticos de grandes dimensões. Este problema não está na funcionalidade do programa em si, mas no fato do método auto-consistente estar considerando a energia dipolar, o que demanda de muito poder de processamento.

Este software será aprimorado em possíveis trabalhos futuros, tendo em vista a adição de novos recursos, de modo que também remova a limitação de recursos que ele tem, assim como evoluir sua estrutura e aumentar a eficiência. O fato deste software ser de código aberto, possibilita que recursos também possam ser adicionados por outros programadores.

REFERÊNCIAS

- AGRA, K. L. **Estudo da Dinâmica de Magnetização em Multicamadas Magnetostrictivas**. Tese de Doutorado. Departamento de Física Teórica e Experimental. UFRN. Natal/RN, 2015.
- AHARONI, A. **Introduction to the Theory of Ferromagnetism**. Clarendon Press, Oxford, 1996.
- CAELUM. **Java e Orientação a Objetos**. Disponível em: <<https://www.caelum.com.br/apostila-java-orientacao-objetos>>. Acesso em: 25/05/2018.
- DANTAS, A. L., et al. *J. Appl. Phys.* 2005, Vol. 97, p. 105-1.
- DANTAS, A. L., CAMLEY, R. E. e CARRICO, A. S.. *J. Appl. Phys.* 2007, Vols. 75, 123907.
- DANTAS, A. L. **Desafios do Nanomagnetismo**. Revista FAPERN, v. 7, p. 144-149, 2008.
- DANTAS, A. L., REBOUÇAS, G. O. G. e CARRIÇO, A. S.. *Journal of Applied Physics*, v. 105, p. 07C116. 2009.
- DANTAS, A. L., REBOUÇAS G. O. G., CARRIÇO A. S.. Washington, DC: s.n., 2010. 11th Joint MMM-Intermag.
- FRAZÃO, C. R.. **Modelo de Ising em Sistemas Núcleo/Casca Nanomagnéticos**. Dissertação de mestrado. Departamento de Ciências de Materiais. UNB. Brasília/DF, 2014.
- GUIMARÃES, A. P.. **Principles of Nanomagnetism**. Berlin: Springer, first edition, 2009.
- Object Oriented MicroMagnetic Framework**. Disponível em: <<http://math.nist.gov/oommf/software.html>>
- REBOUÇAS G. O. G., SILVA A. S. W. T., DANTAS, A. L., CAMLEY R. E., CARRIÇO A. S.. *Physical Review. B*. 2009, Vols. 79, p.104402.
- SILVA, Adan Sady de Medeiros. **Nanoelementos Ferromagnéticos Acoplados a Substratos Antiferromagnético**. Tese de Doutorado. Departamento de Física. UERN. Mossoró/RN, 2011.
- SILVA, A. S. M., DANTAS, Ana L., REBOUÇAS, G. O. G. and CARRIÇO, Artur S.. *Journal of Applied Physics*. 2011, Vols. v.109, p.07D314.
- STONER, E. C.; WOHLFARTH, E. P.. **A mechanism of magnetic hysteresis in heterogeneous alloys**. Philosophical Transactions of the Royal Society A: Physical, Mathematical and Engineering Sciences, 1948. 240 (826): 599–642.
- REBOUÇAS, G. O. G.. **Nucleação de Vórtices e Paredes de Domínio em Nanoestruturas Magnéticas**. Tese de Doutorado. Departamento de Física Teórica e Experimental. UFRN. Natal/RN, 2010.

APÊNDICE A – Atributos e métodos das classes do MagSimu

Nas classes que serão apresentadas a seguir, serão desconsiderados elementos desnecessários como componentes de estrutura que organizam a interface, alguns métodos gerados pelo NetBeans 8.2 que são comuns a muitas aplicações feitas nesse IDE.

A.1. CLASSE “Principal”

A classe de nome “Principal” é a classe principal do programa e gera uma interface gráfica simples com uma área de texto para mostrar o log da execução e um painel para o sistema ser desenhado. Seus atributos e métodos estão representados detalhadamente na Figura A.1.

Figura A.1 – Classe “Principal”

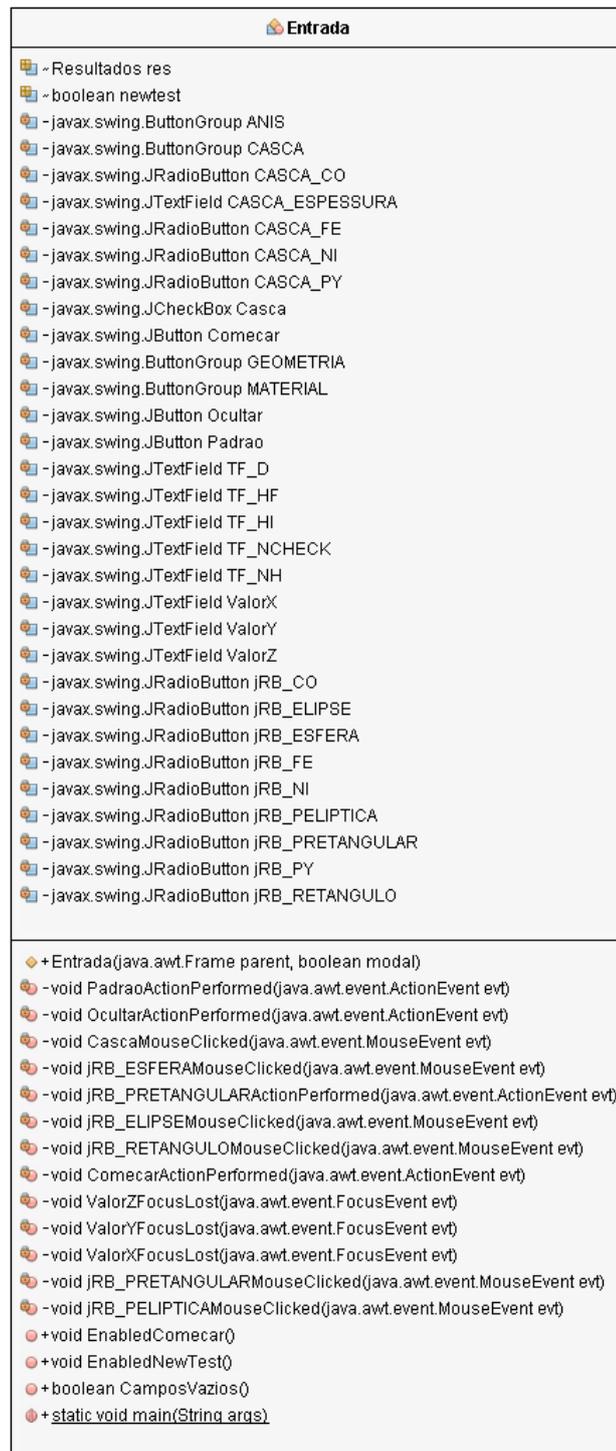


Fonte: Elaborado pelo autor.

A.2. CLASSE “Entrada”

A classe responsável por receber as informações do usuário por uma interface gráfica é a classe “Entrada”. Seus atributos e métodos estão representados detalhadamente na Figura A.2.

Figura A.2 – Classe “Entrada”



Fonte: Elaborado pelo autor.

A.3. CLASSE “JRede”

É a classe “JRede” que fará os mapas de vetores dos momentos magnéticos do sistema no painel branco de “Principal”. Seus atributos e métodos estão representados detalhadamente na Figura A.3.

Figura A.3 – Classe “JRede”

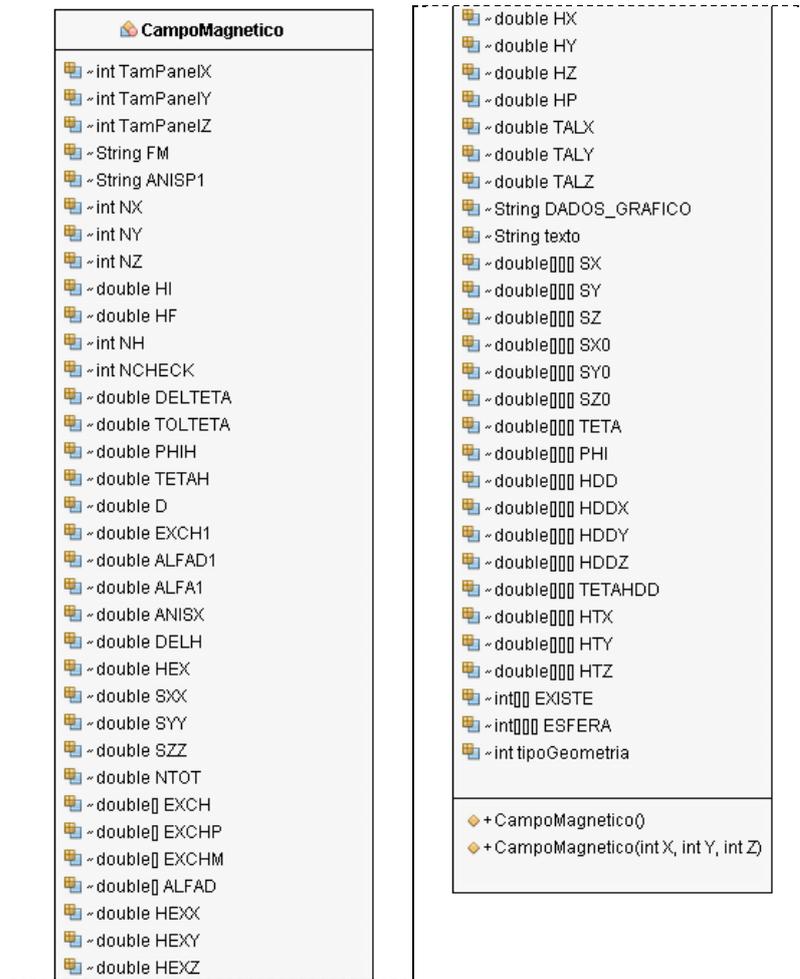


Fonte: Elaborado pelo autor.

A.4. CLASSE “CampoMagnetico”

A classe que passa todos os seus atributos por herança para a classe “Resultados” e as classes que irão calcular as configurações de equilíbrio dos sistemas magnéticos. Seus atributos e métodos estão representados detalhadamente na Figura A.4.

Figura A.4 – Classe “CampoMagnético”

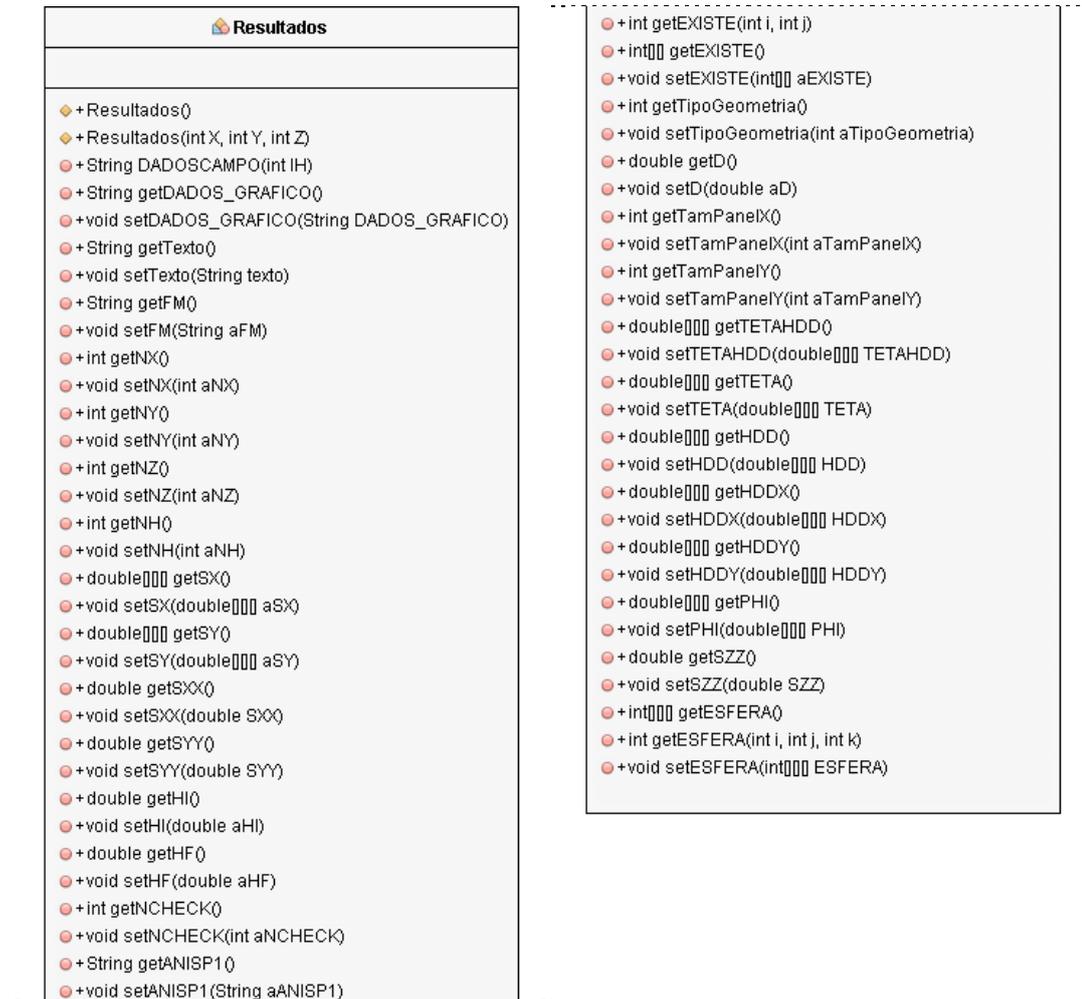


Fonte: Elaborado pelo autor.

A.5. CLASSE “Resultados”

A classe “Resultados” herda todos os atributos de “CampoMagnético”. Ela armazena e reutiliza as informações do usuário e dos processos. Como seus atributos são todos herdados, ele possui nenhum próprio, e os seus métodos estão representados detalhadamente na Figura A.5.

Figura A.5 – Classe “Resultados”

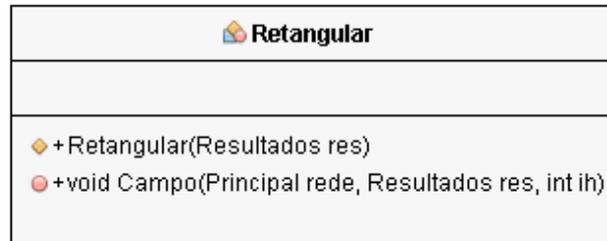


Fonte: Elaborado pelo autor.

A.6. CLASSE “Retangular”

A classe “Retangular” realiza os cálculos da simulação quando a geometria escolhida é a retangular. Como seus atributos são todos herdados, ele possui nenhum próprio, e os seus métodos estão representados detalhadamente na Figura A.6.

Figura A.6 – Classe “Retangular”

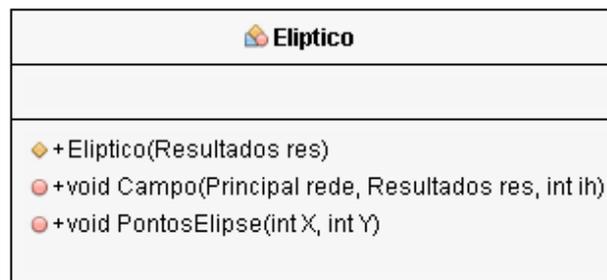


Fonte: Elaborado pelo autor.

A.7. CLASSE “Eliptico”

A classe “Eliptico” realiza os cálculos da simulação quando a geometria escolhida é elíptica. Como seus atributos são todos herdados, ele possui nenhum próprio, e os seus métodos estão representados detalhadamente na Figura A.7.

Figura A.7 – Classe “Eliptico”

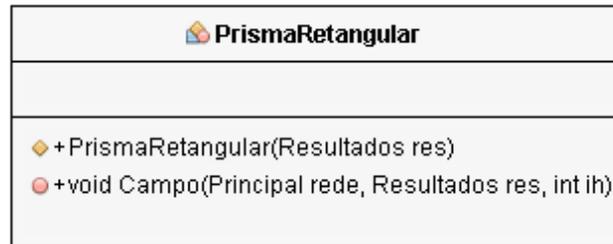


Fonte: Elaborado pelo autor.

A.8. CLASSE “PrismaRetangular”

A classe “PrismaRetangular” realiza os cálculos da simulação quando a geometria escolhida é for um prisma retangular. Como seus atributos são todos herdados, ele possui nenhum próprio, e os seus métodos estão representados detalhadamente na Figura A.8.

Figura A.8 – Classe “PrismaRetangular”

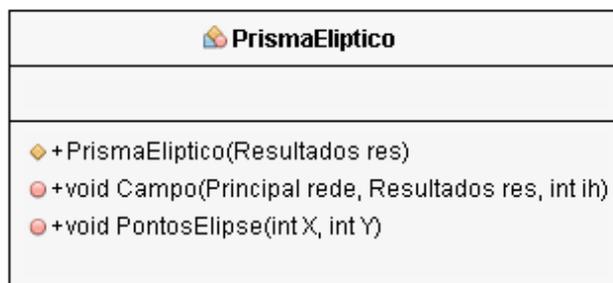


Fonte: Elaborado pelo autor.

A.9. CLASSE “PrismaElptico”

A classe “PrismaElptico” realiza os cálculos da simulação quando a geometria escolhida é for um prisma elíptico. Como seus atributos são todos herdados, ele possui nenhum próprio, e os seus métodos estão representados detalhadamente na Figura A.9.

Figura A.9 – Classe “PrismaElptico”

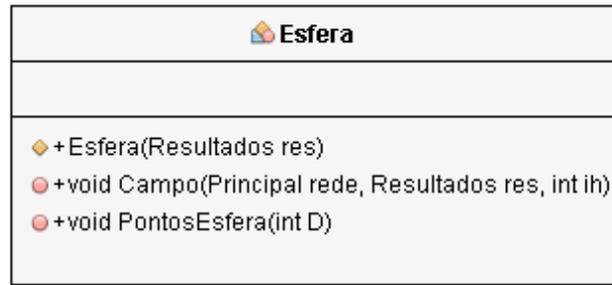


Fonte: Elaborado pelo autor.

A.10. CLASSE “Esfera”

A classe “Esfera” realiza os cálculos da simulação quando a geometria escolhida é for uma esfera. Como seus atributos são todos herdados, ele possui nenhum próprio, e os seus métodos estão representados detalhadamente na Figura A.10.

Figura A.10 – Classe “Esfera”



Fonte: Elaborado pelo autor.