

SIMULAÇÃO DE COMPOSTOS FERROMAGNÉTICOS: CARACTERÍSTICAS PARA QUALIDADE DA FERRAMENTA

Ana Cláudia Souza- anaclaudia@ice.ufjf.br

Roberto Ferreira - roberto.ferreira@ice.ufjf.br

João Paulo Peçanha- joaopaulo@ice.ufjf.br

Fernanda Campos- fernanda.campos@ufjf.edu.br

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

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

Departamento de Ciência da Computação

Sócrates Dantas- dantas@fisica.ufjf.br

Departamento de Física

Universidade Federal de Juiz de Fora, Cidade Universitária,

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

Resumo. *Este trabalho apresenta as características de qualidade identificadas a partir da visão dos usuários para a ferramenta Simulador de Compostos Ferromagnéticos (SINCOFER). O levantamento de atributos do software teve como base o modelo de qualidade descrito pela Norma ISO/NBR 9126-1, do qual foram selecionadas para esse estudo as características de funcionalidade e usabilidade. Foram utilizadas duas métricas para o processo de identificação dos atributos mais importantes do software: média ponderada e uso do decaimento quadrático. Conseguiu-se com este trabalho identificar onze características de qualidade que se fazem necessárias na implementação do simulador.*

Keywords: *Qualidade de Software, Simulação Física, Software de Simulação*

1. Introdução

A área acadêmica necessita de vários instrumentos tecnológicos para facilitar pesquisas e auxiliar nas metodologias de ensino. Podemos citar, como exemplo, a Física da Matéria Condensada que, com sua abordagem teórica, necessita de simuladores para construção e análise dos modelos estudados. Encontrar simuladores que ofereçam esta funcionalidade com um baixo custo se torna um problema que, muitas vezes, impede ou atrasa trabalhos que poderiam resultar em novas descobertas.

Ainda no escopo da Física da Matéria Condensada, o desenvolvimento de uma ferramenta que ofereça ao pesquisador a possibilidade de definir, de modo simples, a forma dos objetos a serem simulados, é de grande interesse. As propriedades eletromagnéticas de tais objetos podem então ser estudadas através do uso de simuladores, levando a um entendimento mais aprofundado das questões teóricas que regem os processos de simulação.

A ferramenta Simulador de Compostos Ferromagnéticos (SINCOFER) foi desenvolvida com intuito de auxiliar os físicos na criação e composição de objetos relevantes para pesquisa em Física da Matéria Condensada. Para leigos, descrever as estruturas a serem simuladas através de ferramentas de CAD é muitas uma tarefa complexa. Em outras situações, levar tal descrição para um simulador constitui-se uma tarefa ainda mais árdua, visto que os simulado-

res atuais não oferecem tal funcionalidade. A ferramenta descrita neste trabalho, por sua vez, facilita a criação, visualização e simulação dessas estruturas.

O desenvolvimento de ferramentas como o SINCOFER é uma tarefa complexa. Sem o criterioso levantamento de requisitos funcionais e de usabilidade, ao final do processo de desenvolvimento pode-se criar uma ferramenta que não atenda as necessidades dos seus utilizadores. Assim, é desejável a aplicação de modelos de qualidade descritos em normas que possam garantir a qualidade final do produto.

Qualidade de software é um tema que aborda todo o processo de desenvolvimento e também o produto de software a ser construído ou adquirido. Para se identificar a qualidade de um produto é necessário saber o que os clientes querem, precisam e esperam do software. Dessa forma o usuário é o árbitro na avaliação da qualidade do produto de software.

Neste trabalho realizamos o levantamento de atributos de qualidade necessárias para o SINCOFER, de modo que o produto final possa atender aos anseios de seus usuários. Com este propósito, utilizamos o modelo de qualidade para características externas e internas definidas pela norma ISO/IEC 9126 (ABNT, 2003). Foram selecionadas as sub-características contidas nas características funcionalidade e usabilidade. Para identificar os atributos de qualidade foram realizadas entrevistas com usuários e participantes do projeto que identificaram e validaram essas características quanto a sua importância para a ferramenta de simulação.

Esse artigo está organizado da seguinte forma. A Seção 2 apresenta características do Simulador de Compostos Ferromagnéticos (SINCOFER) e descreve a importância deste software no meio acadêmico. A Seção 3 descreve qualidade de software segundo a norma ISO/IEC 9126 (ABNT, 2003), dando enfoque maior nas características funcionalidade e usabilidade. Na Seção 4 estão descritos métodos e métricas aplicados para conseguir encontrar atributos de qualidade do SINCOFER. Por fim, na Seção 5, serão apresentadas as conclusões finais deste artigo.

2. Simulador de Compostos Ferromagnéticos

Simulações computacionais de fenômenos físicos geralmente envolvem uma alta complexidade de espaço (recursos computacionais necessários para se realizar a simulação) e de tempo (tempo gasto para realização dos cálculos inerentes à simulação). Portanto, uma estruturação consciente do software se faz necessário para que essa complexidade possa ser tratada de maneira eficiente. Nesta seção descreveremos com detalhes o planejamento e implementação dos módulos que compõem o SINCOFER. A figura 1 mostra o diagrama que representa a organização destes módulos.

Para que sejam alcançados bons resultados numéricos referentes à simulação de um tipo de sistema físico, nesse caso simulação de ferromagnetos, é importante a escolha de modelos físicos e computacionais adequados. Para simular a interação atômica em compostos ferromagnéticos cristalinos foi adotado o modelo de Heisenberg (Terrones et al., 2006). Este modelo é representado por uma equação matemática composta fundamentalmente de uma importante propriedade quântica do átomo: o *spin*. O cálculo da interação entre *spins* de átomos é portanto uma das principais operações a serem realizadas ao longo do processo.

Os átomos podem ser arranjados computacionalmente como pontos fixos no espaço providos de uma carga vetorial não fixa (*spin*). Na natureza, observa-se que diversos fatores fazem com que essa carga tenha seu valor alterado. Portanto, para simular esse comportamento, é necessário um modelo computacional confiável que seja capaz de reproduzir essa interação do elemento com o ambiente e com os outros átomos do sistema. Para tal, foi escolhido o algoritmo de Monte Carlo Metrópolis (Metropolis et al., 1953). Detalhes da implementação desse método podem ser encontrados em (Landau & Binder, 2005).

O modelo físico-matemático previamente citado não apresenta grandes desafios quanto

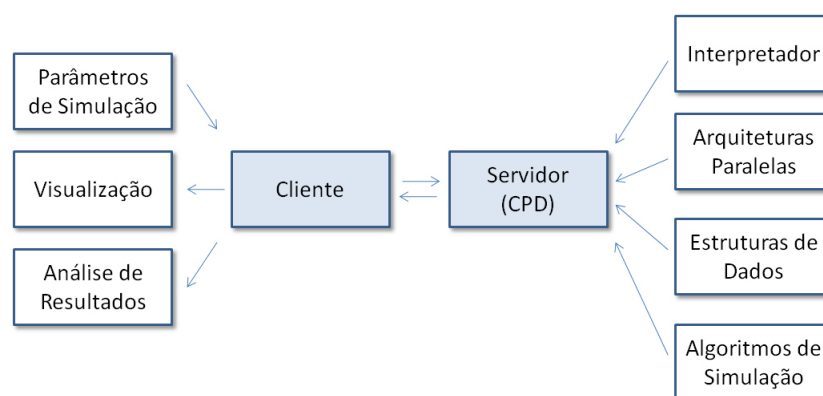


Figura 1: Diagrama que representa esquematicamente a modularização do simulador.

a resolução numérica da equação que o descreve. Contudo, essa operação é realizada $N \times N$ vezes para N átomos presentes no sistema. Esse fator quadrático é considerado, portanto, um problema a ser solucionado computacionalmente. Para aumentar a eficiência do simulador e diminuir o tempo total gasto para a realização desses cálculos, este foi dividido em dois grandes módulos e implementados usando a arquitetura cliente-servidor.

O cliente conta com uma interface gráfica que é usada para comunicação com o servidor. Através dessa interface é possível passar todos os parâmetros de simulação necessários para que essa se inicie: dimensões do sistema, tipo de estrutura de dados a serem utilizadas, constantes da equação, etc. Encontra-se ainda no cliente um visualizador implementado usando a biblioteca gráfica OpenGL (Opengl et al., 2005), que mostra em tempo real a situação do sistema (geometria do objeto, posição espacial dos átomos, vetor de *spin*, partições realizadas no objeto e gráficos com resultados instantâneos).

Neste tipo de arquitetura, o módulo cliente e servidor se localizam em máquinas distintas. Para realizar a comunicação entre os processos, foi usado o paradigma de programação *socket* (Stevens, 1997). Esse paradigma fornece um controle dos dados enviados/recebidos em um nível mais baixo, permitindo com isso, controle total sobre a troca de mensagens. Para realizar a comunicação utilizamos um protocolo da camada de transporte orientado a conexão confiável, que garante a entrega de dados sem erros no fluxo dos bytes: TCP/IP.

Uma importante característica do sistema simulado, e que é de fundamental importância para o servidor, é a forma no qual seus átomos estão dispostos ao longo de sua estrutura. É de interesse de cientistas a modelagem de objeto com as mais variadas formas geométricas. Vale frisar que a experimentação computacional permite a criação de formas inexistentes na natureza, servindo, portanto, como uma ferramenta importantíssima para pesquisa. Com o objetivo de fornecer liberdade na criação desses objetos, foi desenvolvida uma ferramenta que é capaz de gerar os mais variados tipos e formas de estruturas geométricas. A interface para a entrada do código responsável por gerar essas estruturas encontra-se no módulo cliente.

O desenvolvimento desta ferramenta, responsável por gerar objetos geométricos específicos, teve como base a linguagem sugerida em (Ferreira, 2009). Considerando que os usuários da ferramenta seriam notadamente físicos e cientistas da computação a questão da usabilidade teve prioridade no planejamento, bem como a sua manutenção, tendo em vista que a mesma deverá evoluir ao longo do tempo.

A proximidade com linguagens de programação já existentes foi uma estratégia utilizada. Nesse contexto, a linguagem C foi escolhida como principal exemplo para estruturas por ser a

linguagem mais difundida no meio acadêmico. Para facilitar a construção de qualquer expressão, foram inseridas à gramática todas as funções matemáticas contidas em uma biblioteca da linguagem de programação C denominada *cmath*. Todas as estruturas internas e externas da ferramenta foram avaliadas e ajustadas cautelosamente de forma que a ficasse realmente simples de construir e compreender as estruturas.

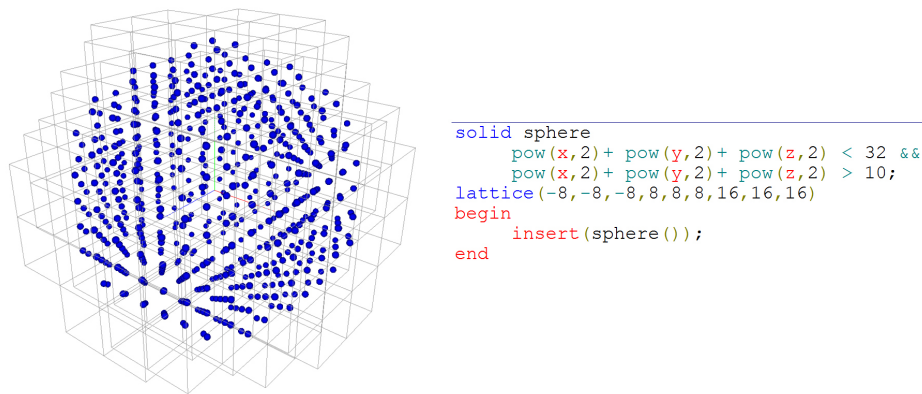


Figura 2: Representação geométrica do objeto e sua respectiva codificação.

Com a linguagem definida por uma gramática pertencente à classe LL(1), o interpretador pode ser construído. Um interpretador é um programa que traduz um código fonte em um código executável. No caso específico, sua função é converter um código inserido pelo usuário, em uma estrutura de dados única (que representa um objeto geométrico) pronta para ser simulada. A Figura 2 ilustra uma esfera gerada a partir de um código escrito usando a linguagem mencionada.

Em relação ao projeto e implementação deste interpretador, podemos evidenciar três etapas bem definidas no que diz respeito à sua construção: analisador léxico, analisador sintático e analisador semântico.

Com a estrutura do objeto pronta, nos resta realizar a comunicação do cliente com o servidor, de fato. As informações referentes aos parâmetros de configuração do sistema navegam pela rede local e/ou global até chegarem ao seu destino: servidor de dados. Esse módulo constitui a espinha dorsal do software, por assim dizer. O servidor, ou Central de Processamento de Dados (CPD), realiza efetivamente todos os cálculos referentes à simulação. A massa de dados estará organizada de forma mais simples (matriz tridimensional), ou de forma mais complexa (árvore de subparticionamento espacial *Octree*), isso dependerá da escolha do usuário. A segunda opção representa a base para o processamento paralelo baseado em múltiplos fluxos de execução existente no software.

Como dito anteriormente, o alto custo computacional para a resolução numérica do modelo físico-matemático nos conduz a buscar caminhos que auxiliem na diminuição do tempo de processamento do sistema. A partir disso, foram desenvolvidos módulos específicos baseados em arquiteturas paralelas: processamento de dados usando placas gráficas de propósito geral (*General-purpose computing on Graphics Processing Units - GPGPU*), usando a tecnologia CUDA (NVIDIA, 2007), e processamento paralelo baseado em múltiplas linhas de execução (*Multithreading*) (Butenhof, 1997).

3. Qualidade de Software

Qualidade de software pode ser vista como um conjunto de características que devem ser alcançadas em um determinado grau para que o produto atenda as necessidades de seus usuários

(Rocha et al., 2001). Para a norma NBR ISO 8402, qualidade é a totalidade das características de uma entidade, que lhe confere a capacidade de satisfazer necessidades explícitas e implícitas [NBR ISO 8402 in Rocha et al. (2001)].

O objetivo, ao se desenvolver um produto de software, não é alcançar a qualidade perfeita, mas sim a qualidade necessária e suficiente para o uso especificado, quando o produto for entregue e realmente utilizado pelos usuários (Rocha et al., 2001).

A qualidade de produto de software é baseada em normas que avaliam se o produto satisfaz o cliente e tem fácil manutenção. É o resultado direto das atividades realizadas no processo de desenvolvimento do software. Alguns exemplos de normas de qualidade de produto são as normas ISO/IEC 14598, 12119 e 9126 (Sodré, 2006).

A satisfação do usuário ou cliente está diretamente relacionada à qualidade e é percebida de formas diferentes. Qualidade de software é um conceito complexo, relativo (depende sempre da perspectiva de quem está avaliando) e jamais se pode pensar em qualidade como sinônimo de perfeição [Lima, 2003].

Segundo a norma ISO/IEC 9126-1 (ABNT, 2003) a qualidade do produto de software pode ser especificada pelas métricas de qualidade em uso, pelas métricas externas e internas. Obter um produto que satisfaça as necessidades do usuário normalmente requer uma abordagem iterativa para o desenvolvimento de software com *feedback* contínuo sob a perspectiva do usuário (ABNT, 2003).

Os requisitos de qualidade externa especificam o nível de qualidade requerido sob o ponto de vista externo. Eles incluem requisitos derivados das necessidades de qualidade dos usuários, incluindo os requisitos de qualidade em uso. Os requisitos de qualidade externa são usados como meta para validação em vários estágios de desenvolvimento.

Os requisitos de qualidade interna especificam o nível de qualidade requerido sob o ponto de vista interno do produto. Os requisitos de qualidade interna são usados para especificar as propriedades dos produtos intermediários. Estes podem incluir modelos estáticos e dinâmicos, outros documentos e código-fonte. Requisitos de qualidade interna podem ser usados como metas para validação em vários estágios de desenvolvimento. Eles também podem ser usados para definir estratégias de desenvolvimento e critérios de avaliação e de verificação durante o desenvolvimento.

Qualidade em uso é a visão da qualidade do produto de software do ponto de vista do usuário, quando este produto é usado em um ambiente e um contexto de uso especificados. Ela mede o quanto usuários podem atingir seus objetivos num determinado ambiente e não as propriedades do software em si (ABNT, 2003).

A Norma NBR ISO/IEC 9126 categoriza os atributos de qualidade de software em seis características (funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade) as quais são, por sua vez, subdivididas em sub-características como pode ser visto na Figura 3. As sub-características podem ser medidas por meio de métricas externas e internas.

Este trabalho teve o foco em apenas duas características: funcionalidade e usabilidade, que serão detalhas a seguir.

3.1 Funcionalidade

Define-se funcionalidade por capacidade do produto de software de prover funções que atendam às necessidades explícitas e implícitas, quando o software estiver sendo utilizado sob condições especificadas (ABNT, 2003). Ela é sub-dividida nas seguintes sub-características:

- Adequação: capacidade do produto de software de prover um conjunto apropriado de funções para tarefas e objetivos do usuário especificados.

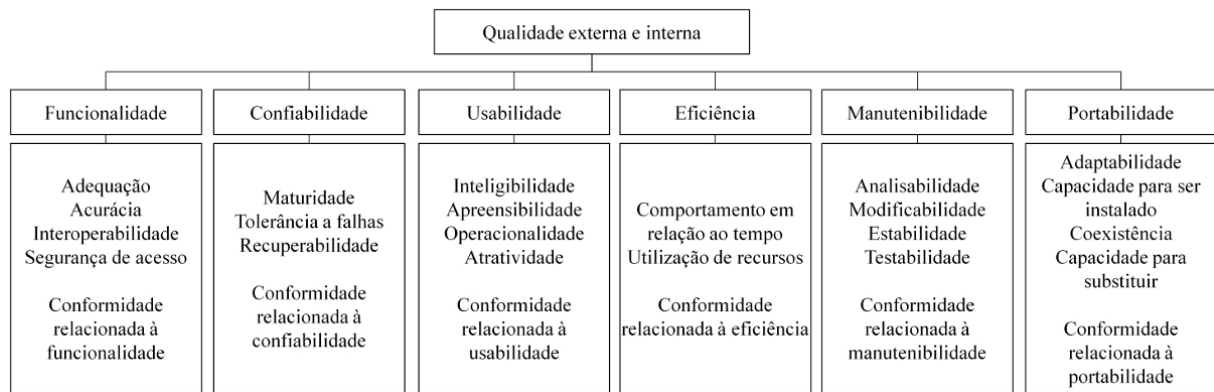


Figura 3: Modelo de qualidade para qualidade externa e interna.

- **Acurácia:** capacidade do produto de software de prover, com o grau de precisão necessário, resultados ou efeitos corretos ou conforme acordados.
- **Interoperabilidade:** capacidade do produto de software de interagir com um ou mais sistemas especificados.
- **Segurança de acesso:** capacidade do produto de software de proteger informações e dados, de forma que pessoas ou sistemas não autorizados não possam lê-los nem modificá-los e que não seja negado o acesso às pessoas ou sistemas autorizados.

Conformidade relacionada à funcionalidade: capacidade do produto de software de estar de acordo com normas, convenções ou regulamentações previstas em leis e prescrições similares relacionadas à funcionalidade.

3.2 Usabilidade

Define-se usabilidade por capacidade do produto de software de ser compreendido, aprendido, operado e atraente ao usuário, quando usado sob condições especificadas (ABNT, 2003).

Ela é sub-dividida nas seguintes sub-características:

- **Inteligibilidade:** capacidade do produto de software de possibilitar ao usuário compreender se o software é apropriado e como ele pode ser usado para tarefas e condições de uso específicas.
- **Apreensibilidade:** capacidade do produto de software de possibilitar ao usuário aprender sua aplicação.
- **Operacionalidade:** capacidade do produto de software de possibilitar ao usuário operá-lo e controlá-lo.
- **Atratividade:** capacidade do produto de software de ser atraente ao usuário.
- **Conformidade relacionada à usabilidade:** capacidade do produto de software de estar de acordo com normas, convenções, guias de estilo ou regulamentações relacionadas à usabilidade.

Para se obter a qualidade desejada de produtos de software, são necessários modelos que viabilizem a avaliação da qualidade desses produtos. O principal propósito da mensuração da qualidade de software é fornecer resultados quantitativos referentes aos produtos de software

para que estes sejam compreensíveis, aceitáveis e confiáveis por qualquer parte interessada (ISO, 1987).

4. Qualidade do SINCOFER

A qualidade do produto final do Gerador de Estruturas Ferromagnéticas foi uma preocupação constante dos pesquisadores e dos desenvolvedores envolvidos no projeto. Foi definido, então, que seriam identificados os atributos de qualidade do produto com foco nas características de funcionalidade e usabilidade da Norma ISO/IEC 9126 (ABNT, 2003), em um processo que envolveu seis atores, citados na Tabela 1, que são potenciais usuários do software. Com a identificação destas características espera-se também que todo o processo de desenvolvimento garanta que o produto final do software atenderá a esse conjunto de critérios de qualidade.

Ator	Título
Ator 1	D. Sc. em Ciência da Computação
Ator 2	D. em Eng. de Sistemas e Computação
Ator 3	Doutor em Ciências
Ator 4	Graduando em Ciência da Computação
Ator 5	Graduando em Ciência da Computação
Ator 6	Mestrando em Modelagem Computacional

Tabela 1: Atores Envolvidos.

Nas entrevistas iniciais, para levantamento das características de qualidade, participaram os quatro primeiros atores descritos na Tabela 1. Considerando que os atores envolvidos não conheciam as características e sub-características do modelo de qualidade da norma ISO/IEC 9126 (ABNT, 2003), foi realizada uma etapa de apresentação da mesma. Na segunda etapa, cada entrevistado sugeriu características de qualidade que considerava importantes para o software gerador de estruturas ferromagnéticas. Todos os entrevistados concordaram que a sub-característica segurança de dados não se aplica ao software, o que poderá ser avaliado no futuro é a utilização de criptografia com chaves públicas nos dados de saída.

As entrevistas foram realizadas de forma individual, garantindo o sigilo das propostas. Este fato foi importante, pois permitiu identificar os pontos em comum e o que cada um considera importante e prioritário. Neste levantamento foram identificados 45 atributos de qualidade.

Tendo como base as entrevistas realizadas e os atributos sugeridos, foi construída uma ficha de validação para que os seis participantes do processo selecionassem, de acordo com o seu julgamento, o grau de importância da característica listada, numa escala de três opções: muito importante, importante ou sem importância. Todas as notas foram contabilizadas, e para realizar a seleção dos atributos mais importantes, foi feito um comparativo entre duas métricas: média ponderada e uso do decaimento quadrático.

A primeira métrica selecionada para selecionar as características mais importantes para os atores foi a média ponderada de cada atributo avaliado. Portanto, cada nota "Muito importante", "Importante" e "Sem importância" recebeu os seguintes pesos 4, 2 e 1, respectivamente. Estes pesos foram selecionados de forma que os artefatos considerados mais importantes fossem priorizados. Após aplicar esta métrica em todos os pontos, obteve-se o gráfico apresentado na Figura 4.

Com esta métrica obteve-se a classificação dos vinte primeiros atributos com maiores pontuações. As maiores notas foram selecionadas porque são as candidatas a atributos do software, por terem sido consideradas mais importantes que as outras. A segunda métrica selecionada para identificar as características mais importantes para os atores utilizou-se o decaimento qua-

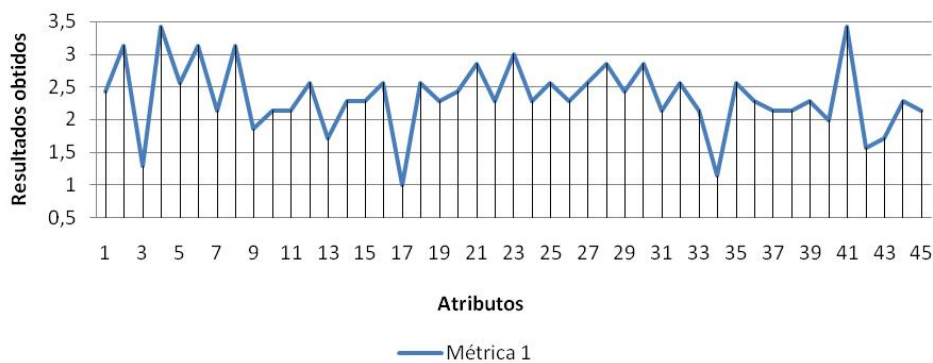


Figura 4: Resultados da aplicação da métrica 1.

drástico das notas para "Sem importância". Com isto fica fácil visualizar quais atributos são mais importantes e quais possuem a maior probabilidade de sair da lista. Após aplicar esta métrica em todos os pontos, obteve-se o gráfico apresentado na Figura 5.

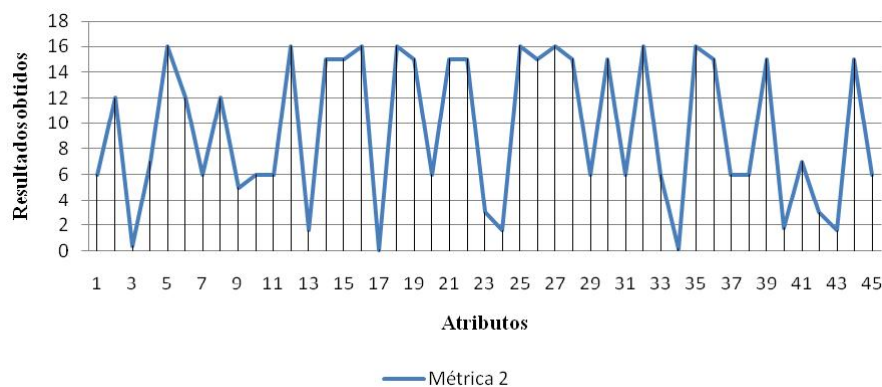


Figura 5: Resultados da aplicação da métrica 2.

Para esta métrica também foram extraídos os vinte primeiros atributos classificados com maiores pontuações. Da mesma forma que a métrica anterior, as maiores notas foram selecionadas porque são as candidatas a atributos do software, por terem sido consideradas mais importantes que as outras.

Após o processo de identificação e validação das características de qualidade para o produto de software gerador de estruturas ferromagnéticas, onde se priorizou as características de funcionalidade e usabilidade, a partir da Norma ISO/IEC 9126, foram então destacados os atributos descritos na Tabela 2.

Para chegar ao resultado das principais características foi feita uma interseção entre os resultados das duas métricas aplicadas que resultou em onze atributos. Portanto, estes atributos podem ser considerados os principais atributos de qualidade que devem constar do software ao término da construção do mesmo.

Com estes atributos será possível realizar a avaliação do produto de software ao longo do processo de desenvolvimento e após o término de sua construção.

5. Considerações Finais

Alcançar a qualidade do produto de software não é algo trivial. É necessário que todas as etapas de um processo de desenvolvimento passem por uma avaliação para assim garantir a

Sub-característica	Atributos de Funcionalidade
Adequação	Ter ambiente para geração de objetos ferromagnéticos arbitrários.
Adequação	Ter uma linguagem capaz de gerar todas as estruturas possíveis.
Interoperabilidade	Ter interface de entrada textual, mas com possibilidade de passagem remota do programa.
Interoperabilidade	Ter uma saída de forma que outros softwares consigam ler.
Interoperabilidade	Interagir com o Monte Carlo <i>spins</i> em tempo de utilização. (aceitando suas entradas e passando corretamente suas estruturas de saídas)
Conformidade	O compilador deve estar em conformidade com a linguagem sugerida.
Sub-característica	Atributos de Usabilidade
Inteligibilidade	Ter uma linguagem fácil e próxima a linguagem comuns.
Inteligibilidade	Ter uma documentação de apresentação para que usuários saibam o que existe no software.
Inteligibilidade	Ter o código bem documentado para facilitar futuras expansões, manutenção e integrações.
Apreensibilidade	Ter mensagens de erros intuitivas.
Operacionalidade	Deve sempre retornar ou um erro ou a estrutura.

Tabela 2: Atributos de qualidade do gerador de estrutura.

qualidade final do produto.

Fazer o levantamento de requisitos requer envolvimento de todos os usuários e *stakeholders* do projeto. Portanto, existe uma enorme necessidade de realizar reuniões, analisar os dados, realizar pesquisas e entrevistas para conseguir chegar a um ponto em que o produto de software saia com os resultados esperados por todos.

Com os requisitos em mãos, o processo de software deve estar ou retornar ao início do processo de construção do software para que sejam incluídos todos os itens necessários que garantam a qualidade do produto atendendo aos atributos levantados pelos usuários.

Para trabalhos futuros pretendemos fazer o levantamento de atributos do software gerador de estruturas ferromagnéticas nas demais características do modelo de qualidade da Norma 9126-1 (ABNT, 2003).

REFERÊNCIAS

- ABNT, A. B. D. N. T., 2003. Nbr iso/iec 9126 tecnologia de informação: Avaliação de produto de software, características de qualidade e diretrizes para o seu uso. Technical report, Rio de Janeiro.
- Butenhof, D. R., 1997. *Programming with POSIX Threads*. Addison-Wesley Reading, MA, USA, 1 edition.
- Ferreira, R. C., 2009. Linguagem para geração de objetos implícitos para simulação de spins. Technical report, Universidade Federal de Juiz de Fora, Juiz de Fora.
- ISO, T. I. O. F. S., 1987. Iso 9000-9003: Quality management systems. Technical report, United Kingdom.
- Landau, D. & Binder, K., 2005. *A Guide to Monte Carlo Simulations in Statistical Physics*. Cambridge University Press, New York, NY, USA.
- 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.

- Opengl, Shreiner, D., Woo, M., Neider, J., & Davis, T., 2005. *OpenGL(R) Programming Guide : The Official Guide to Learning OpenGL(R), Version 2 (5th Edition)*. Addison-Wesley Professional.
- Rocha, A. R. C., Maldonado, J. C., & Weber, K. C., 2001. *Qualidade de Software: Teoria e Prática*. Pearson Education, São Paulo.
- Sodré, C. C. P., 2006. Norma iso/iec 9126: Avaliação de qualidade de produtos de software. Technical report, Universidade Estadual de Londrina, Londrina.
- Stevens, W. R., 1997. *UNIX Network Programming: Networking APIs: Sockets and XTI*. Prentice Hall PTR, Upper Saddle River, NJ, USA.
- Terrones, H., López-Urías, F., Muñoz-Sandoval, E., Rodríguez-Manzo, J., Zamudio, A., Elías, A., & Terrones, M., 2006. Magnetism in fe-based and carbon nanostructures: Theory and applications. *Solid State Sciences*, vol. 8, n. 3-4, pp. 303 – 320. Special issue in honour of A.K.Cheetham.

SIMULATION OF FERROMAGNETIC COMPOUNDS: CHARACTERISTICS FOR THE TOOL QUALITY

Abstract. *This paper aims to present the quality characteristics for the Ferromagnetic Compounds Simulator (SINCOFER), identified from the user viewpoint. The characteristics identification was based on the quality model described in ISO/NBR 9126-1 norm, of which functionality and usability characteristics were selected for this study. Two metrics were used for the more important software attributes identification process: weighted average and quadratic decay. Through this work it was possible to identify eleven quality characteristics that are important for the simulator.*

Keywords: *Software quality, physics simulation, simulation software*