

Um Aplicativo Educacional para Elaboração e Simulação Dinâmica de Diagramas - Abordagem Utilizando Filas

*An Educational Application for Elaboration and Dynamic Simulation of Diagrams - One Approach
Using Queues*

Nicolas Marcelli Bontempo Vespucio^{1*}, José Maurício Schneedorf Ferreira da Silva¹, Luiz Eduardo da Silva¹

¹ Universidade Federal de Alfenas - Unifal-MG

*e-mail: nick.bontempo@bcc.unifal-mg.edu.br

Abstract

The traditional teaching of Biochemistry, and more specifically its metabolic maps, is generally considered a complex and abstract matter by the students, due to the several relationships presents among its molecules. Although theses maps lead to a better organization and lesser complexity about that interactions, they are almost present as static and printed images, without its natural flow and forces involved. Towards to alleviate the mind-boggling comprehension of metabolic maps, this work present an educational software named SISMA - Autocatalytic Maps System, which was developed using Java programming language, due to its versatility, efficiency, platform portability, and security. The software allows to construct, modify and analyse metabolic maps, and to view them dynamically, with the emergence of products simultaneously to the disappearance of their reagents. The overall implementations in SISMA reflects its potential use as an educational tool for enhance teaching and learning in the discipline of Biochemistry courses by its users.

Keywords: Biochemistry; Software; Metabolism;

Resumo

O ensino tradicional de Bioquímica, e mais especificamente, seus mapas metabólicos, geralmente é considerado um assunto complexo e abstrato pelos alunos, devido aos vários relacionamentos presentes entre suas moléculas. Embora esses mapas conduzam a uma melhor organização e menor complexidade sobre essas interações, estão quase presentes como imagens estáticas e impressas, sem o fluxo natural entre as forças envolvidas. Com o objetivo de facilitar a compreensão dos mapas metabólicos, este trabalho apresenta um software educacional chamado SISMA - Sistema de Mapas Autocatalíticos, que foi desenvolvido usando a linguagem de programação Java, devido à sua versatilidade, eficiência, portabilidade da plataforma e segurança. O software permite construir, modificar e analisar mapas metabólicos e visualizá-los dinamicamente, através do surgimento de produtos simultaneamente ao desaparecimento de seus reagentes. As implementações do SISMA refletem seu uso potencial como uma ferramenta educacional para aprimorar o ensino e a aprendizagem na disciplina de cursos de Bioquímica por seus usuários.

Palavras-chave: Bioquímica; Software; Metabolismo;

1 Introdução

No atual contexto social, de grandes avanços tecnológicos, o acesso amplo a informação e percepção do mundo vivo como uma rede de relações dinâmicas e em constante transformação, tem-se discutido a necessidade de urgentes mudanças nas instituições de ensino visando, entre outros aspectos, à reconstrução de seu papel social [1].

O processo de ensino-aprendizagem, deve ser analisado em toda a sua complexidade, para que não seja tratado de maneira simplista e restrita, e fique centrado somente na reprodução do conhecimento [2, 3]. Acompanhando esse contexto, a educação superior, passa por uma mudança de paradigma, afastando-se de uma metodologia tradicional de ensino, centrada no professor para processos de aprendizagem focados nos estudantes, configurando as chamadas metodologias ativas de ensino-aprendizagem [4].

Diante do avanço tecnológico existente, estabeleceu-se uma geração de estudantes visuais, que cresceram em meio à interatividade inerente a ambientes digitais e atualização de dispositivos eletrônicos, e que apresentam, em geral, aversão aos métodos tradicionais de ensino, além de um baixíssimo limiar para assistir com conforto às aulas meramente expositivas [5, 6].

Desta forma, o ensino superior tem incorporado em seu desenvolvimento o uso de metodologias ativas, de estratégias de resolução de problemas e a utilização de novas tecnologias de informática e telecomunicação. Os software educativos são um exemplo dessa estratégia, por apresentar a grande capacidade de despertar a curiosidade e o interesse do estudante sobre conteúdos abordados, tornam o ensino mais participativo e a aprendizagem mais significativa.

Assim, professores e estudantes encontram nesses software uma ferramenta poderosa para conferir uma nova dimensão ao conhecimento, facilitando a aprendizagem de temas complexos, possibilitando que o conteúdo seja apresentado por meio de esquemas interativos, estáticos e animados, que ilustram e facilitam a visualização [6].

O objetivo deste trabalho foi desenvolver um software, capaz de elucidar conceitos da Bioquímica e tornar mais claro os conceitos tratados em sala de aula, através da simulação de mapas autocatalíticos.

2 Ensino da bioquímica

Bioquímica é a ciência que estuda as moléculas (substâncias) que constituem os seres vivos e as transformações que ocorrem entre essas.

Seu conteúdo é particularmente difícil, pois apresenta conceitos complexos, que exigem conhecimento prévio. Muitos discentes não conseguem acompanhar uma aula expositiva e apresentam dificuldade no aprendizado, pois grande parte dos fenômenos estudados, ocorrem no interior das células, em nível microscópico e, portanto, pouco observáveis e abstratos [7].

A dificuldade no ensino-aprendizagem dos conteúdos da disciplina, decorrentes dos elementos desafiadores apontados, tanto da disciplina, quanto do novo perfil do discente, podem ser contornados por meio de formas alternativas, flexíveis, com múltiplas formas de representação e graus de interatividade, que facilitem a abstração necessária à compreensão do tema [8].

A Bioquímica tal como vista em sala de aula e nos livros didáticos é, normalmente dividida em dois grandes blocos sequenciais: a estrutura e função das biomoléculas, e sua multiplicidade de interconversões químicas. A essa última dá-se o nome de metabolismo e, dada a complexidade numérica e de inter-relações das biomoléculas constituintes das células vivas, costuma ser organizada e sumarizada em livros-texto por mapas metabólicos [9].

Neste sentido, um mapa metabólico tende a facilitar o processo ensino-aprendizagem, uma vez que reúne as informações de biomoléculas e de suas interações, como apresentado na Figura 1 abaixo.

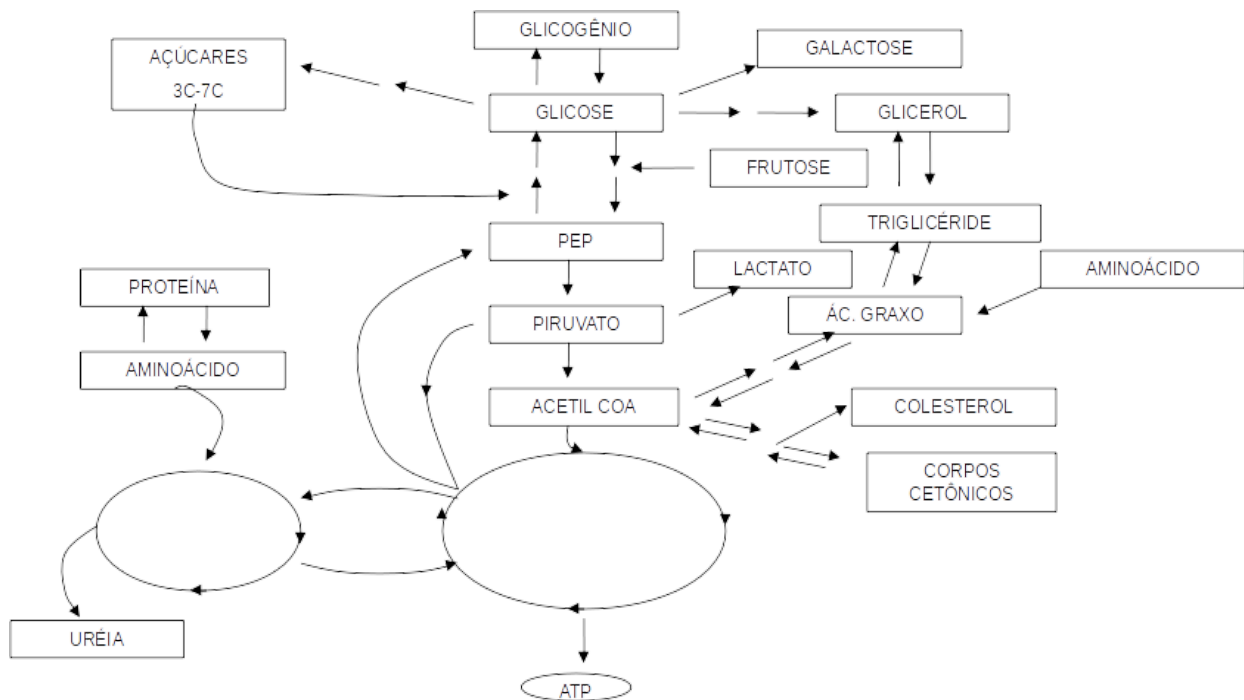


Figura 1. Exemplo de mapa metabólico simplificado

Apesar dessa facilitação aparente, os mapas metabólicos são construídos de forma estática, não permitindo ao aprendiz a visualização das transformações que ocorrem entre reagentes e produtos, o que torna complexa sua compreensão. Assim, embora o metabolismo seja naturalmente concebido como um equilíbrio de forças (moléculas e suas quantidades) e de fluxos (transformações das moléculas) [10], os mapas metabólicos apresentam somente as forças.

Dessa forma teve-se a ideia de desenvolver um software que representasse o comportamento dinâmico dos mapas metabólicos, fornecendo uma retratação mais dinâmica do funcionamento do metabolismo expresso nesses, o SISMA.

3 SISMA

O software SISMA - acrônimo para Sistema de Mapas Autocatalíticos foi desenvolvido com o objetivo de tornar mais simples o estudo de mapas metabólicos por

estudantes da Bioquímica, utilizando-se das vantagens intrínsecas da visualização dinâmica e de possibilidades de interação do aluno com o objeto em estudo.

O SISMA então foi desenvolvido com o objetivo de tornar a representação de mapas metabólicos mais compreensível para os alunos, com a possibilidade de simular os seus comportamentos. Como resultado final foi concebido um software que tem funcionalidades abrangentes, capaz de simular as reações metabólicas propostas por seu usuário, bem como qualquer outro tipo de mapa em que ocorra transformações (processos autocatalíticos).

O software em sua primeira versão foi elaborado por um grupo transdisciplinar da Ciência da Computação e da Bioquímica da UNIFAL-MG, coautores deste trabalho, e depositado no Instituto de Propriedade Industrial - INPI em 19/05/2008 sob o registro no. 0000270802591357. Atualmente possui aplicabilidade educacional, sendo utilizado por docentes na UNIFAL-MG como ferramenta auxiliar para visualização e compreensão de mapas metabólicos.

Na primeira tela do software, o usuário evidencia várias funcionalidades, tais como a possibilidade de abertura de mapas previamente salvos, criação e salvamento do mapa em elaboração, a introdução de objetos de sua quantidade relativa (as forças, no caso, biomoléculas), a introdução de setas conectando os objetos criados (os fluxos, no caso, os caminhos metabólicos de conversão de substratos em produtos), a inserção de imagens para facilitação do aprendizado, além de outras funções evidenciadas pela Figura 2. As imagens nos ícones e a ajuda por texto melhoram a usabilidade do software evidenciado pela Figura 3, oferecendo acesso as funcionalidades que são mais simples de serem compreendidas.

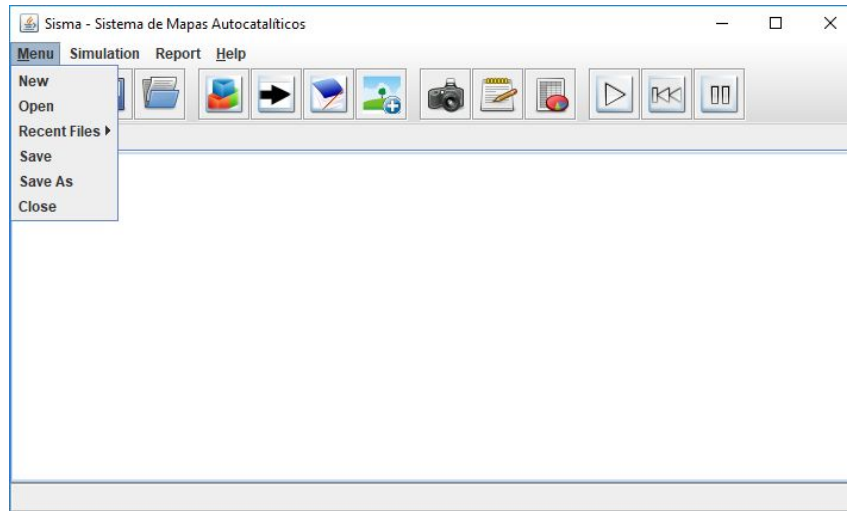


Figura 2. Tela Inicial do Software com menu aberto

O software foi concebido de forma a permitir a seu usuário a construção de mapas simples ou complexos tendo em mente o que o mesmo faria se estivesse com um lápis ou caneta, e uma folha de papel em branco. Essa formatação objetivou simplificar ao máximo sua usabilidade, ao mesmo tempo em que permitisse a inserção, remoção e edição de informações na própria tela sob elaboração, valendo-se apenas de cliques de mouse sobre os itens nela contidos (objetos, setas). Durante a criação de objetos na tela, uma funcionalidade significativa do software consistiu no reposicionamento de objetos na mesma por um simples clique-arraste do mouse.

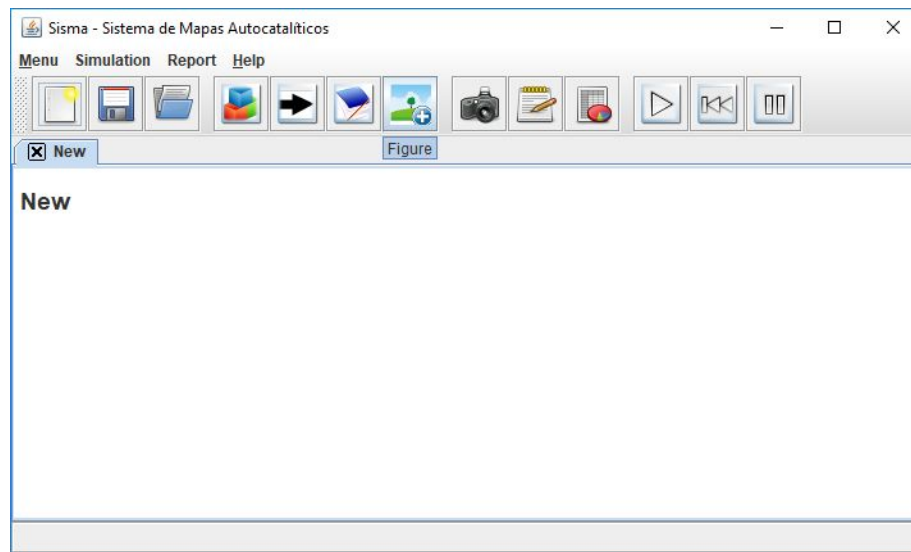


Figura 3. Tela Inicial do Software com demonstração da clareza nos botões

Para criação de um mapa, deve-se inicialmente posicionar o mouse no ponto desejado da tela e, com um clique desse, inserir as informações do objeto (nome, cor, intensidade relativa). Essa intensidade relativa percorrendo de zero a cem unidades arbitrárias, constitui a intensidade da matiz que atribuída ao objeto, e pode ser ajustada com uma barra de rolagem na tela específica de cada objeto criado (Figura 4). As caixas de marcação exemplificadas na Figura 4 permitem o cômputo do objeto para a elaboração de um gráfico de simulação em tempo real mas arbitrário ("Plot"), e sua inserção optativa ao início da simulação pretendida ("Init"). As Figuras 5 e 6 exemplificam respectivamente a edição dos objetos e dos caminhos.

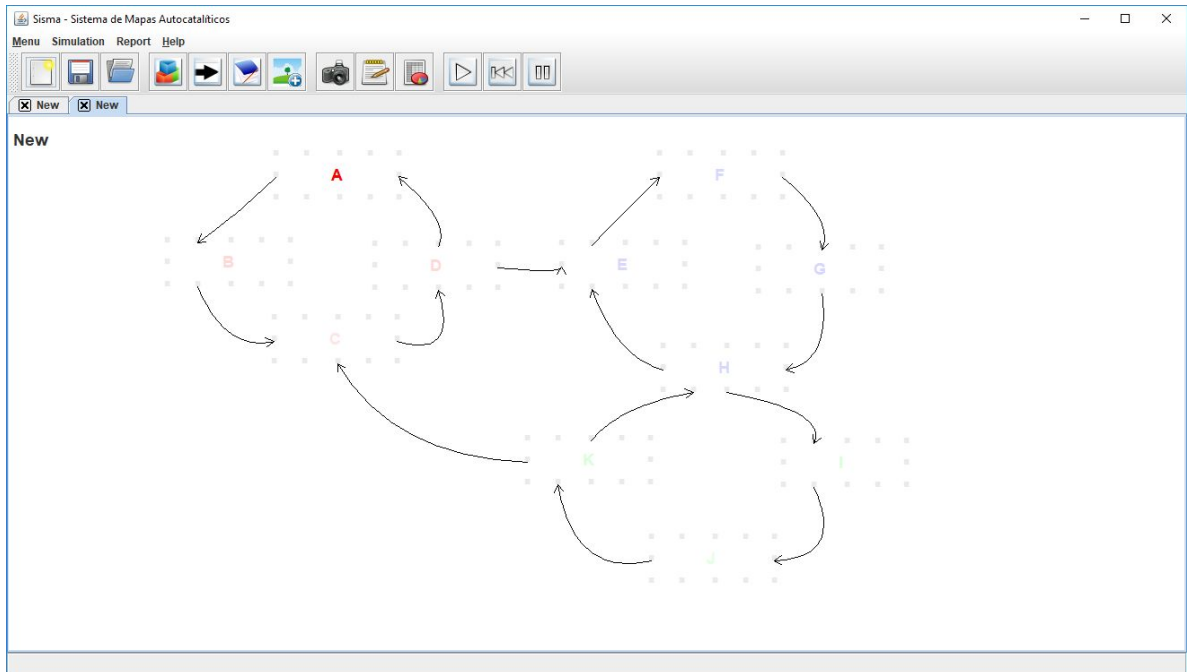


Figura 4. Mapa de reações com 11 objetos e 14 caminhos.

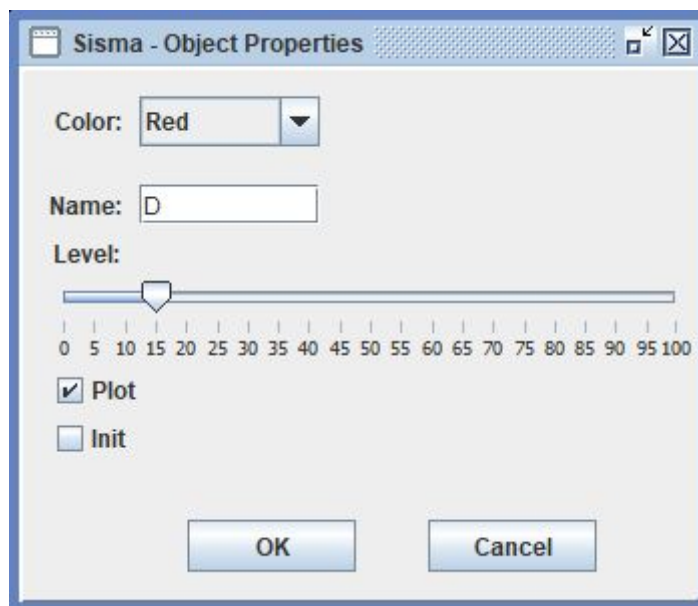


Figura 5. Propriedades do objeto.

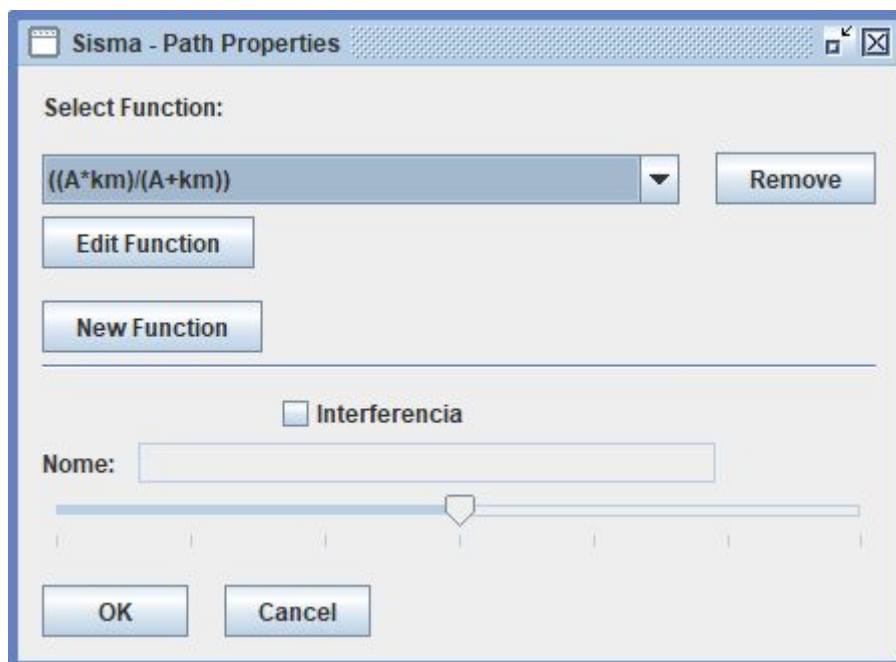


Figura 6. Propriedades do caminho.

A simulação pode ser facilmente acompanhada e as interações dos objetos são claras e graduais, conforme os valores são transferidos em razão da função especificada pelo usuário na via. Dessa forma, durante a transformação de um reagente em produto, as matizes de cor dos reagentes vão ficando mais claras até se tornarem invisíveis, ao passo que a dos produtos passam de uma visualização tênue àquela demarcada pela função matemática definida pelo usuário para a transformação, representando a transferência completa (Figura 7).

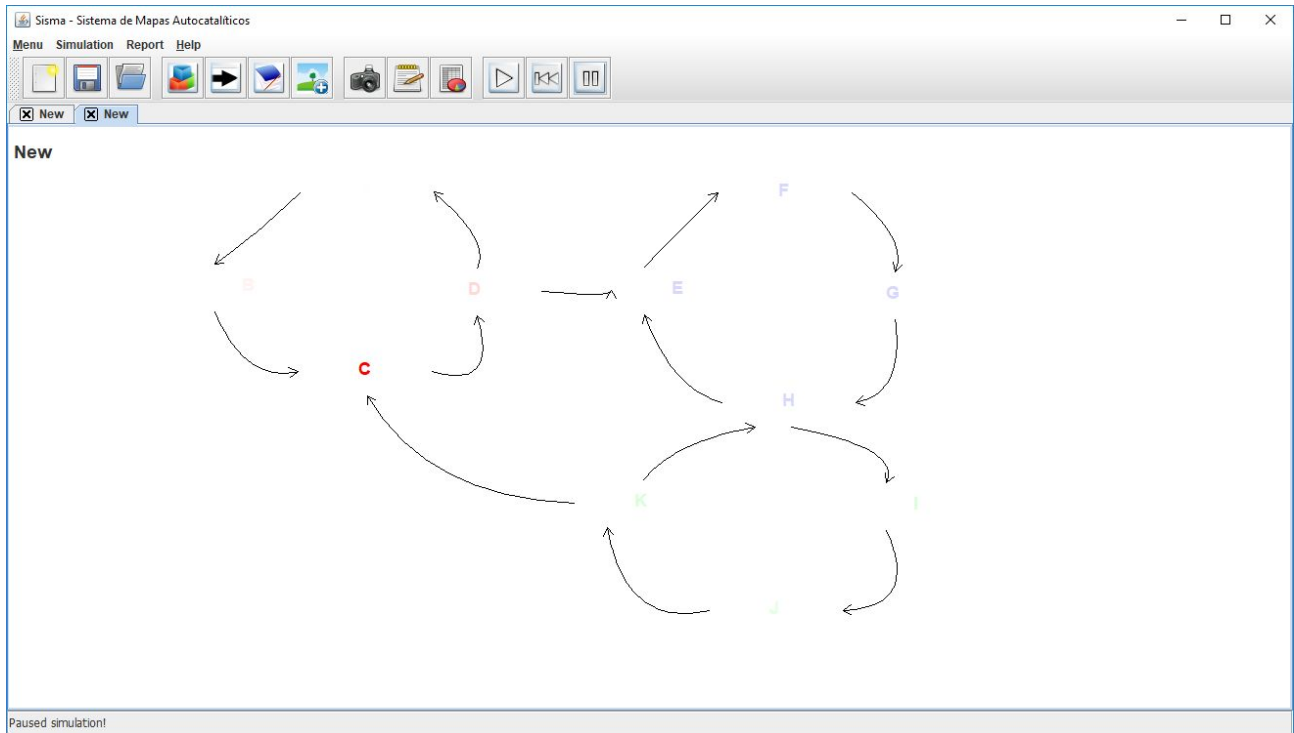


Figura 7. Simulação, o objeto A já está invisível e o B está transparente, assim o C recebe os valores desses dois objetos e continua com o andamento da simulação.

4 Implementação

O software foi escrito na linguagem de programação *Java* e distribuído como um pacote *JAR (Java ARchive)*. Para o desenvolvimento da interface de usuário foi utilizado a *framework Swing*, facilitando o trabalho de criação de uma interface com usabilidade. Uma característica da escolha do *Java* foi a possibilidade de execução multiplataforma, tendo sido testado nas plataformas *Windows* e *Linux*.

Para sua implementação foi seguido os princípios do padrão de desenvolvimento de software *DDD (Domain-Driven-Design)*, ocasionando um software com camadas, que podem ser facilmente combinadas e reusadas, o que também garante sua fácil manutenção e melhoria [11].

A divisão do software estruturou-se nas seguintes camadas: apresentação, responsável pela interface do usuário; domínio, encarregada pelos modelos; aplicação,

responsável pelas lógicas do domínio; e por fim, a camada de infra estrutura que ficou incumbida da persistência dos dados na interação com o arquivo externo.

Como o objetivo do software é representar fluxos e reações complexas, um dos desafios do desenvolvimento é a simulação por eventos discretos (SED), já que os estados mudam de valor em pontos discretos no tempo [12]. Com isso foi implementada a abordagem por três fases (ABC), a qual apenas os tempos de execução contados são aqueles que possuem eventos no sistema [13]. Nessa abordagem temos a primeira fase (A) que verifica a próxima atividade a ser executada, a segunda fase (B) que executa os eventos incondicionais do momento e a terceira fase (C) que executa os eventos condicionais [13]. Dessa forma, executando a simulação de forma mais rápida, já que não se torna necessário simular momentos de tempo em que não ocorre nenhum evento.

A versão atual do software utiliza a estrutura de filas (*FIFO - First in First out*) para o controle da execução, o qual os primeiros objetos são analisados e se possuem vias de transferência de valores para outros objetos, eles são adicionados para a fila de execução no próximo tempo discreto. Assim são formados os tempos discretos da simulação, nos quais podem ocorrer mais de uma transferência de valores no mesmo momento, que são os eventos da simulação do modelo.

As fórmulas implementadas para a transferência de valores entre os objetos são implementadas utilizando conceitos de compiladores, primeiramente a análise léxica que remove entradas não utilizadas para a geração do código final, nesse caso a remoção de espaços em branco da entrada do usuário [14]. As análises sintática e semântica são importantes para verificar se a fórmula é válida, podendo reconhecer se alguma variável não reconhecida foi inserida na fórmula, tendo a possibilidade de ser uma constante, então com esse reconhecimento o software possibilita que seja inserido um valor a essa constante [14]. Por fim é gerado o código intermediário em uma pilha que é utilizada para a transferência de valores [15]. Essa funcionalidade reconhece fórmulas com operações como soma, subtração, multiplicação, divisão, logaritmo, exponencial e raiz.

A representação do mapa pode ser facilmente traduzida em um XML (*eXtensible Markup Language*), que é uma linguagem de marcação utilizada na web para

transferência de dados, ela apresenta sintaxe simples, assim como certa facilidade para a leitura [16]. Por esses motivos foi a escolha para a funcionalidade de salvamento de arquivos, dessa forma as representações dos dados são repassadas facilmente entre diferentes dispositivos, já que são contidos em apenas um arquivo facilmente transferível.

4 Conclusão

O software foi desenvolvido para tornar a simulação de mapas metabólicos fácil e rápida. Em direção a esse objetivo foi desenvolvido uma interface de usuário simples e objetiva, provendo ao usuário diversas ferramentas para customização do mapa e criando possibilidades diversas em seu uso. Os mapas criados pela ferramenta podem ser facilmente compartilhados entre os interessados, para facilitar discussões e aprofundamento das representações recriadas. Com isso facilitando a visualização das abstrações criadas pelos mapas em software, observando seus comportamentos dinâmicos e tornando melhor o entendimento em sala de aula.

Em síntese, o presente trabalho buscou um aprimoramento efetivo para o SISMA, através da refatoração do software utilizando a estrutura de filas para o controle de execução, assim criando um código mais simples e previsível, melhorando a manutenibilidade do software e aumentando a facilidade para encontrar possíveis erros, se comparado ao anterior através de paralelismo. Com a refatoração teve também mudanças na base de código para torná-lo mais legível e organizado como a reestruturação seguindo o paradigma DDD.

Além de diversas melhorias em usabilidade e aplicabilidade do software para o processo de ensino-aprendizagem, tais como a) Aumento no número de operadores para criação de fórmulas como logaritmo neperiano, raiz, exponencial. b) Aumento no número de conectores dos objetos. c) Salvar sem perguntar, caso queira dar continuidade em um projeto previamente salvo. d) Funcionalidade Salvar como permitindo salvar em locais diferentes do arquivo original. e) Salvar imagem dentro do xml, evitando problemas em referenciar a mesma e não existir, assim não carregando a imagem.

Referencias

- [1] Oliveira MCSAC, Carvalho AF, Junqueira JRA, Furlanetto PG. Estratégias Ativas de Aprendizagem e o Desenvolvimento de Competências Técnicas e Atitudinais. *Ensaio Pioneiros* 2017; 1(1): 139-152.
- [2] Scatigno A, Torres BB. Estilos de aprendizagem em universitários. Tese de doutorado[Psicologia] - Universidade Federal de Campinas. 2016.
- [3] Mitre SM, Siqueira-Batista R, Girardi-de-Mendonça JM, Morais-Pinto NM, Meirelles CAB, Pinto-Porto C, et al. Metodologias ativas de ensino-aprendizagem na formação profissional em saúde: debates atuais. *Ciência & saúde coletiva* 2008; 13(2): 2133-2144.
- [4] Berbel NAN. As metodologias ativas e a promoção da autonomia de estudantes. *Semina: Ciências Sociais e Humanas*. 2011; 32(1): 25-40.
- [5] Thompson P. The digital natives as learners: technology use patterns and approaches to learning. *Computers & Education*. 2013; 65(1): 12-33.
- [6] Garzón JCV, Magrini ML, Costa C, Galembeck E. Realidade aumentada no ensino de vias metabólicas. *Revista de Ensino de Bioquímica*. 2014. 12(2): 129-143.
- [7] Siqueira AJS, Remião JOR, Azevedo AM. Dynamic metabolic diagrams in biochemistry. *Biochemistry and Molecular Biology Education*. 1992. 20(2): 97-98.
- [8] Azevedo AMP, Lazzarotto GB, Timm MI, Zaro MA. Relato de uma experiência com o uso do Diagrama Metabólico Dinâmico Virtual do Ciclo de Krebs. *Novas tecnologias na educação* 2004. 2(1) : 1-9.
- [9] Nelson DL, Albert LL, Michael MC. *Lehninger principles of biochemistry*. 7. ed. London: Macmillan, 2017;
- [10] Prigogine I, Grégoire N. Biological order, structure and instabilities. *Quarterly Reviews of Biophysics*. 1971. 4(2): 107-148.
- [11] Evans E. *Domain-driven design: tackling complexity in the heart of software*. Addison-Wesley Professional. 1. ed. 2003.
- [12] Levy AY, Iwasaki Y, Fikes R. (1997). Automated model selection for simulation based on relevance reasoning. *Artificial Intelligence*. 1997. 96(2): 351-394.
- [13] Pidd M. *Computer Simulation in Management Science*. Addison-Wesley Professional. 5. ed. Inglaterra: Chichester, 2004.
- [14] Loudon KC. *Compiladores-Princípios e Práticas*. Cengage Learning Editores. 1. ed. Portugal, 2005.
- [15] Cormen TH, Leiserson CE, Rivest RL, Stein C. *Algoritmos: teoria e prática*. Editora Campus. 3. ed. Estados Unidos: 2002.
- [16] Almeida, MB. Uma introdução ao XML, sua utilização na Internet e alguns conceitos complementares. *Ciência da informação*. 2002. 31(2): 5-13.