

9º CONGRESSO BRASILEIRO DE PESQUISA E DESENVOLVIMENTO EM PETRÓLEO E GÁS



9º CONGRESSO
Brasileiro de P&D em
PETRÓLEO E GÁS

Maceió, AL
de 09 a 11 de novembro
2017

Realização:



ABPG
ASSOCIAÇÃO BRASILEIRA DE P&D EM
PETRÓLEO E GÁS



TÍTULO DO TRABALHO:

UM SISTEMA GRÁFICO INTERATIVO PARA ANÁLISE DE REDES DE ESCOAMENTO DE GÁS NATURAL

AUTORES:

Kim Rocha Gama, Ricardo Albuquerque Fernandes, Diogo Tenório Cintra, Adeildo Soares Ramos Júnior, Eduardo Setton Sampaio da Silveira.

INSTITUIÇÃO:

Laboratório de Computação Científica, Centro de Tecnologia, Universidade Federal de Alagoas

Este Trabalho foi preparado para apresentação no 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Petróleo e Gás - 9º PDPETRO, realizado pela Associação Brasileira de P&D em Petróleo e Gás - ABPG, no período de 09 a 11 de novembro de 2017, em Maceió/AL. Esse Trabalho foi selecionado pelo Comitê Científico do evento para apresentação, seguindo as informações contidas no documento submetido pelo(s) autor(es). O conteúdo do Trabalho, como apresentado, não foi revisado pela ABPG. Os organizadores não irão traduzir ou corrigir os textos recebidos. O material conforme, apresentado, não necessariamente reflete as opiniões da Associação Brasileira de P&D em Petróleo e Gás. O(s) autor(es) tem conhecimento e aprovação de que este Trabalho seja publicado nos Anais do 9º PDPETRO.

UM SISTEMA GRÁFICO INTERATIVO PARA ANÁLISE DE REDES DE ESCOAMENTO DE GÁS NATURAL

Abstract

In the actual business scenario, the generation of data related to production and monitoring of demands and services is increasingly common. The majority of natural gas distributors in Brazil works with buried pipelines that are self-connected, resulting in branched meshes and making their analysis and monitoring very complex tasks, demanding computational methods to attend these needs. In the present study, it is proposed the development of a graphic interactive system to analyze pipeflow networks of natural gas. This integrated tool presents a graphic user interface and is integrated with a in-house numerical simulator for pipeflow equations. Input and edit data functionalities work through a database, using various unit systems and precision control of decimal digits configured by user. Then, it is possible to simulate changes in the mesh and evaluate its behavior, automatically generating new reports that can be used to project network expansions or to help future decision-making processes. Considering the amount of data to be analyzed, the system is developed using C++ computational language, to offer a good performance associated with object-oriented paradigm, which facilitates reusability and code expansion. For the development of graphic user interface, it was used the framework Qt and for data manipulation, the SQLite library. For the visualization of georeferenced models, we use the collaborative project OpenStreetMap, to obtain and display maps using tiles, associated to the OpenGL API for drawing pipeline models. In order to validate the developed tool, case studies of simple distribution systems are shown, illustrating flow rate, pressures and pressure drops along the pipes using colormaps and graphs.

Introdução

Desde a década de 1980, a tecnologia vem avançando mais fortemente e a era digital já é uma realidade em diversas áreas do conhecimento. Na engenharia, por exemplo, a quantidade de dados e sistemas de monitoramento remoto ganha destaque desde o momento em que o homem percebeu que poderia reduzir custos associados ao trabalho humano como também garantir a segurança ocupacional dos seus empregados. Com essa nova forma de monitorar, uma grande quantidade de dados é gerada, favorecendo análises numéricas e estatísticas dessas informações que podem conduzir a resultados de grande importância dentro do ambiente corporativo. Indicadores reais de qualidade operacional e de saúde financeira da empresa podem ser gerados, obtendo-se informações cruciais a partir desses dados e auxiliando na tomada de novas decisões.

Dentro da área de Gás Natural, o desenvolvimento de novas tecnologias segue o rumo natural desse processo. As informações técnicas e operacionais, geralmente obtidas no campo, são essenciais para o monitoramento de processos. Quando, por exemplo, na distribuição e transporte, se consegue os dados de vazão, carga hidráulica e conectividade é possível utilizar de modelos físicos que tratam esses dados e preveem o comportamento durante situações operacionais normais ou até em cenários adversos. Essas informações, então, são fundamentais para a tomada de decisão em situações emergenciais ou ainda auxiliando durante as etapas de projetos, manutenção e ampliação da malha de fluxo.

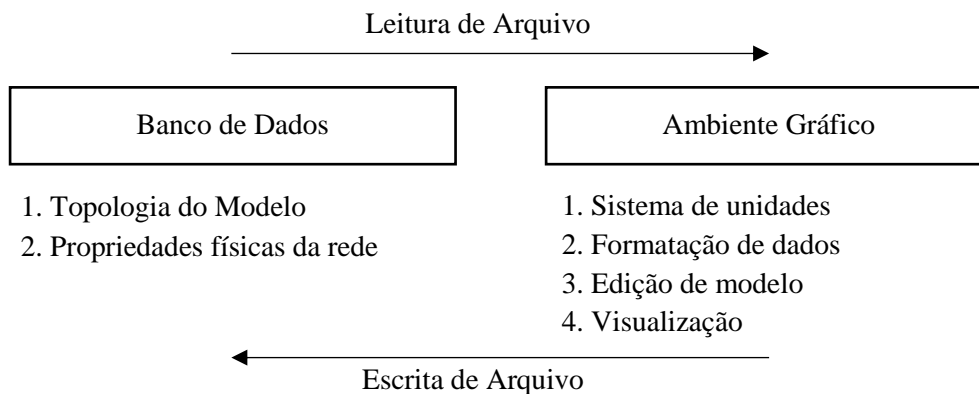
Para facilitar o uso da tecnologia, *softwares* computacionais são desenvolvidos para utilização pelo corpo técnico da empresa. Essas ferramentas viabilizam a modelagem dos problemas em alto nível, geralmente, através de interfaces gráficas com o usuário. No cenário de distribuição e transporte de gás natural, essas interfaces devem oferecer funcionalidades que permitam o usuário carregar e visualizar modelos, de modo que melhor lhe exponha o processo físico, além de conseguir observar os dados referenciados geograficamente, fazer edições, simular os métodos físicos, e obter perspectivas do comportamento obtido para a rede de distribuição. Todas essas características foram inseridas na

interface criada, com o objetivo de facilitar a interação com o usuário, tornando mais eficiente, o processo de tomada de decisões.

Metodologia

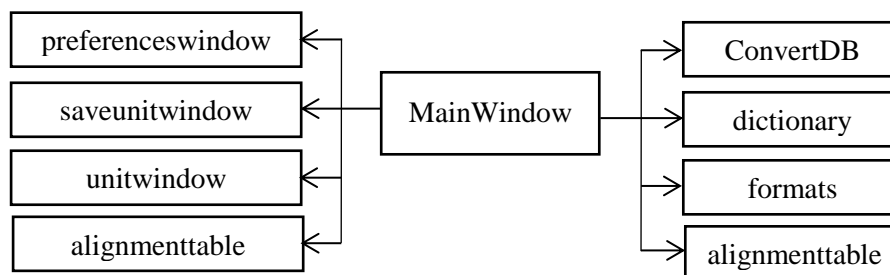
A idealização do MAPFLOW, a interface gráfica apresentada nesse trabalho, possibilita que o usuário faça as alterações necessárias em um modelo de malha de fluxo. A partir da interação com o ambiente gráfico, é possível alterar as informações operacionais e visualizar como se apresenta o atual estado da rede, através do acesso a um banco de dados, onde são armazenadas todas as informações relacionadas à topologia e às propriedades dos dutos, conforme mostra a Fig. 1.

Figura 1: Funcionamento do sistema.



De forma a realizar todo esse processo, é utilizada a linguagem de programação C++ devido a sua ampla eficiência e os aspectos positivos inerentes ao paradigma de programação orientada a objetos (Ellis & Stroustrup, 1993). A arquitetura de classes da interface gráfica do sistema MAPFLOW é mostrada de maneira simplificada na Fig. 2. A classe principal, *MainWindow*, inicializa todas as outras classes relativas a funcionalidades do software. As demais classes e funcionalidades serão discutidas nas seções posteriores. Toda representação gráfica apresentada nesta seção se utiliza do padrão UML (Rumbaugh *et al.*, 2000), o qual define a especificação, visualização e construção de artefatos de sistemas de software.

Figura 2: Arquitetura de classes simplificada.



A *MainWindow* contém todas as funcionalidades básicas do MAPFLOW. Em seu conteúdo, estabelecem-se todas as *callbacks* das ferramentas da janela principal. Assim, dentre suas tarefas básicas, compreende-se o gerenciamento de exposição de dados e de sua busca por alguma informação específica, que é essencial para que as interações entre o usuário e a interface gráfica sejam corretamente

interpretadas e executadas. Essa classe também gerencia qual a propriedade física que será mostrada na tabela, através da sua interligação com a classe myglwidget.

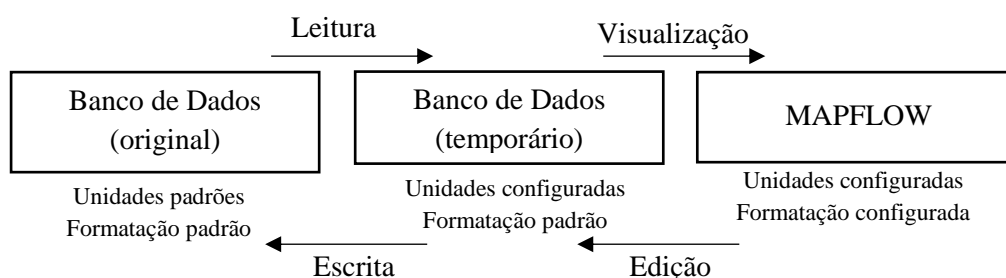
Tendo em vista a necessidade de definir um padrão básico para o banco de dados, a Tabela 1 apresenta o padrão de tabelas que é lido pelo MAPFLOW. Nesse mesmo banco de dados estão os dados do modelo, isto é, os parâmetros necessários para que o simulador consiga analisar matematicamente o modelo e também os respectivos resultados, caso existam.

Tabela 1: Descrição das tabelas do banco de dados.

	Tabela	Conteúdo
Modelo	gas_constituents	Composição das misturas gasosas
	global_settings	Propriedades globais e de convergência
	nodal_prescribed_flow	Prescrições nodais de vazão
	nodal_prescribed_pressure	Prescrições nodais de pressão
	nodes	Definição de nós e suas propriedades
	pipes	Definição de dutos e suas propriedades
Resultado	composition_results	Composições de saída dos nós entregadores
	pipe_results	Resultados da simulação para os dutos
	nodal_results	Resultados da simulação para os nós

A leitura do banco de dados é idealizada através de uma abordagem que possibilite ao usuário visualizar e buscar os dados contidos no arquivo a partir de uma configuração de unidades definida, que pode não ser a mesma configurada pelo simulador. Para viabilizar esse procedimento, como mostra a Figura 3, edições no modelo são realizadas em uma cópia temporária do banco de dados. Nesse processo, o método SetConversionAtTemporaryDb, contido na classe ConvertDB, converte todos os dados ao sistema de unidades escolhido pelo usuário. Isso possibilita o manuseio e interpretação dos dados da forma da forma proposta.

Figura 3: Manipulação do banco de dados para leitura e escrita das informações.



As preferências de usuário que gerenciam as configurações de exibição de dados (unidades e formatação) são tarefas das classes unitwindow e saveunitwindow. O usuário pode usar um sistema previamente definido, como o Sistema Métrico de Unidades ou o Sistema Inglês de Unidade (US), ou ainda alguma convenção da unidades totalmente configurada pelo usuário. O usuário também pode definir a quantidade de casas decimais e a forma com que cada tipo de variável será exibida (formatação científica ou decimal).

Uma vez carregado, as informações do modelo são visualizadas através de uma tabela, utilizando a classe QTableWidgetItem na MainWindow. Na classe myglwidget, a rede é ilustrada através de coordenadas georreferenciadas, desenhada através de primitivas do sistema gráfico OpenGL (Neider

et al, 2003). Nessa classe, utiliza-se ainda um conjunto de mapas (tiles) como plano de fundo, através do OpenStreetMaps.

A MainWindow também é responsável pela edição dos modelos. Essa edição é realizada também na QTableWidgetItem, onde cada dado modificado é salvo no banco de dados temporário ainda com as unidades configuradas pelo usuário. Uma das preocupações existentes nesse processo é a de que não seja perdida a precisão numérica em razão da escolha do número de casas decimais e formato para exibição das variáveis pelo usuário. Dessa forma, ao se tentar editar uma célula da tabela de propriedades, a quantidade nessa célula perde as características de formatação e é apresentada com o máximo de precisão possível, armazenada no banco de dados temporário.

A edição de parâmetros globais, isto é, os parâmetros que são utilizados por todo o modelo e aqueles relativos à convergência do simulador são de responsabilidade da classe preferenceswindow. Essa classe tem como tarefa básica abrir um diálogo que permite que essas alterações sejam feitas. Nota-se, portanto, que a comunicação entre o banco de dados e a os atributos da classe MainWindow também é realizada em outros pontos do *software*.

São dois os caminhos para salvar as alterações. O primeiro, e mais básico, consiste na *callback* relativa ao botão com a função de salvar. O segundo, caso o modelo esteja pronto para ser executado, quando o simulador é chamado, o arquivo é salvo. Esse processo escreve as informações atualizadas do modelo no banco de dados original, salvando as alterações de acordo com as configurações de unidades e formatação padrão (interpretada pelo simulador).

Resultados e Discussão

A visualização de uma rede de dutos interligados responsável por realizar o transporte de gás natural é o principal objetivo do *software* MAPFLOW. Quando captadas as informações de campo, basta que elas abasteçam um modelo válido e estejam na formatação definida pelo padrão de leitura do *software*. Dessa forma, é possível visualizar todo comportamento físico do sistema, fornecendo ao a capacidade de interpretação mais clara do que está acontecendo no campo e possibilitando assim outras perspectivas que possam trazer benefícios em decisões futuras.

O sistema é desenvolvido em C++, já que a implementação de outras funcionalidades na aplicação deve ser plano de projetos futuros e a programação orientada a objetos permite que essa expansão seja possível com relativa facilidade (Ellis & Stroustrup, 1993). Para construção da interface, utilizou-se o *framework* Qt (Arnt, 2015) por ser multi-plataforma e devido a sua facilidade de implementação e simplicidade de seus métodos. Para a manipulação dos dados, trabalha-se com a biblioteca SQLite, devido a sua facilidade de manuseio e principalmente da sua capacidade de filtros de dados e integração com a *framework* Qt. O projeto colaborativo OpenStreetMap foi usado para obter e exibir mapas através de tiles diretamente do OpenGL (Neider et al, 2003). Assim, o MAPFLOW é uma interligação de várias plataformas, sistemas e linguagens em que se possibilita a consolidação de um *software* único com diversas funcionalidades.

O módulo principal da interface compreende as principais zonas que estão indicadas na Fig. 4.

1. Barra de ferramentas. Responsável pelas tarefas básicas de gerenciamento de arquivos. Possibilita ao usuário criar um novo modelo ou abrir um modelo existente e salvá-lo. Também tem como funcionalidade iniciar a simulação do modelo em exibição.

2. Zona de visualização do banco de dados. Permite ao usuário visualizar o conteúdo do banco de dados carregado e escolher uma tabela a ser exibida, dado que, cada tabela apresenta dados referentes

a diferentes entidades do modelo. É dividido em dois blocos, o de modelo, que compreende as tabelas referentes ao modelo físico, e o bloco de resultados, que compreende as tabelas referentes aos resultados (caso existam).

Figura 4: Zonas do ambiente gráfico.



3. Zona de busca. Quando algum modelo estiver carregado, essa zona é responsável por filtrar na tabela corrente, os dados referentes à busca. Esse procedimento é útil quando o modelo compreender muitos dados e o usuário quiser que só sejam exibidas as linhas referentes a alguma propriedade específica.

4. Tabela de dados. Tabela responsável por exibir as informações do modelo carregado. Uma vez exibidas as informações, o usuário pode editar célula por célula, do modo a que lhe for conveniente. A maioria dos recursos de exibição e edição das informações do modelo estão apresentados nessa zona.

5. Barra de ferramentas do mapa. Quando o modelo é exibido no mapa, as linhas que representam dutos podem se colorir, tornando-se um diagrama de cores. Essas cores indicam a intensidade de alguma propriedade física. Essa barra de ferramentas permite ao usuário escolher qual a propriedade física (pressão, vazão, número de Reynolds, fator de fricção, entre outros) que será utilizada. Se o modelo apresentado for quase-estático, também é possível escolher qual o instante de tempo que está sendo visualizado.

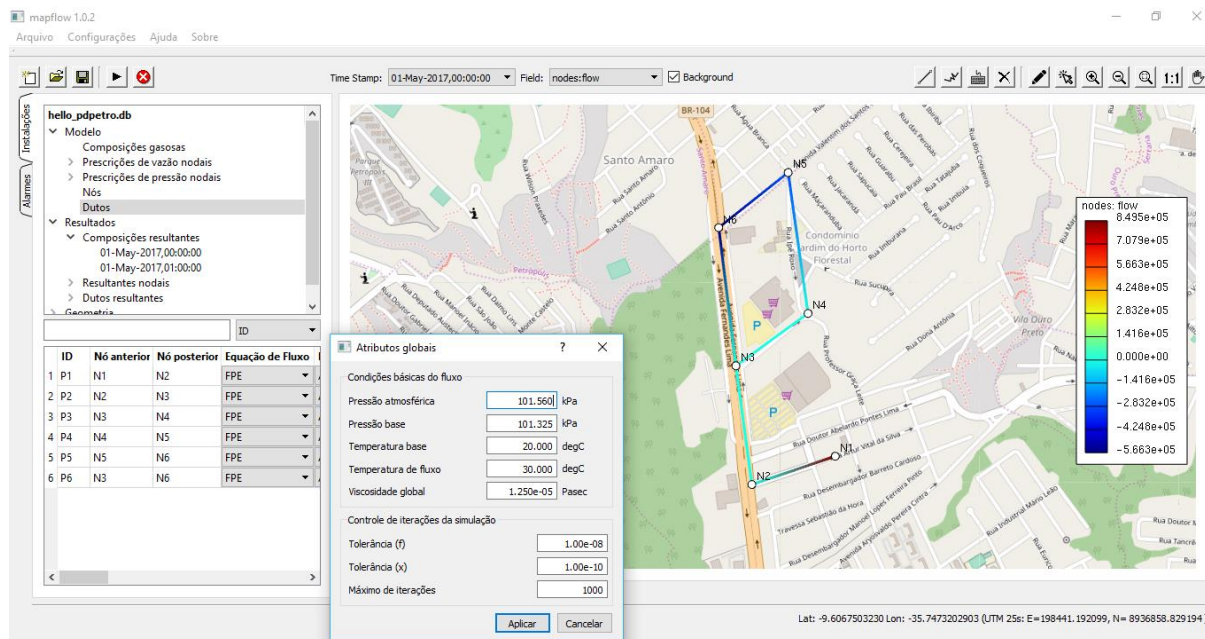
6. Barra de ferramentas de desenho. Uma das formas de se inserir novos dutos ao modelo é diretamente através do mapa. Essa barra de ferramentas possibilita ao usuário a criação de nós e dutos.

7. Display do usuário LatLong. Espaço onde algumas informações são mostradas ao usuário. No estado padrão, são mostradas as coordenadas geográficas em dois sistemas geográficos, o sistema da latitude e longitude e o sistema UTM.

8. Zona de visualização de mapas. Canvas onde é apresentado o modelo físico da rede. Os mapas são exibidos por tiles em background. Os elementos do modelo são ilustrado por linhas indicando os dutos e pontos, que indicam os nós.

Todas essas funcionalidades criam a aplicação mostrada na Figura 5, ilustrando o módulo principal da interface e o diálogo que possibilita a edição de parâmetros globais do modelo.

Figura 5: Aplicação em execução



Conclusões

Este trabalho apresenta o desenvolvimento de um ambiente computacional direcionado à visualização dos problemas de transporte e distribuição de gás natural. O sistema reúne a aplicabilidade e a simplicidade de outras aplicações. O framework Qt é essencial para garantir uma facilidade em criar novos diálogos e a otimizá-los. O uso da biblioteca SQLite para comunicação com banco de dados facilita esse processo, tornando-o rápido e eficiente, entretanto, se o seu uso for aplicado a um banco de dados demasiadamente grande, a aplicação pode perder desempenho, comprometendo a experiência do usuário. No processo de visualização, a utilização do sistema gráfico OpenGL permite o desenho da geometria representativa do modelo de dutos, agregando eficiência na comunicação do sistema gráfico com o banco de dados.

A reunião dos aspectos apresentados nesse trabalho, torna possível a concepção de um ambiente gráfico com os instrumentos necessários para uma interpretação técnica dos resultados de uma maneira satisfatória, possibilitando a tomada de decisões mais assertivas por parte do usuário e garantindo uma boa funcionalidade da aplicação.

Agradecimentos

Os autores desse trabalho agradecem a Algás – Gás de Alagoas S.A pelo financiamento desse projeto e pelo apoio no desenvolvimento do software MAPFLOW.

Referências Bibliográficas

Arnt, A. Al. 2015. Aprendendo Qt com o projeto Octopi. Editora Amazon.

Ellis, M. A., & Stroustrup, B., 1993. C++ Manual de Referência Comentado. Editora Campus.

Neider, J., Davis, T., & Woo, M., 2003. OpenGL Programming Guide (The Red Book). Version 1.3, Addison.

Rumbaugh, J., Jacobson, I., & Booch, G., 2000. *UML: Guia do Usuário*. Editora Campus.