

**UNIVERSIDADE FEDERAL DO TRIÂNGULO MINEIRO
INSTITUTO DE CIÊNCIAS TECNOLÓGICAS E EXATAS
MESTRADO PROFISSIONAL EM INOVAÇÃO TECNOLÓGICA**

GUSTAVO MARINO BOTTA

**IMPLANTAÇÃO DO MODELO DINÂMICO INERCIAL RÍGIDO EM SISTEMAS
HIDRÁULICOS NO SOFTWARE EPANET**

UBERABA-MG

2014

GUSTAVO MARINO BOTTA

**IMPLANTAÇÃO DO MODELO DINÂMICO INERCIAL RÍGIDO EM SISTEMAS
HIDRÁULICOS NO SOFTWARE EPANET**

Dissertação apresentada ao Programa de Mestrado Profissional em Inovação Tecnológica da Universidade Federal do Triângulo Mineiro – UFTM, como requisito parcial para obtenção do título de Mestre em Inovação Tecnológica.

Orientador: Prof. Dr. Luiz Fernando Resende dos Santos Anjo.

UBERABA-MG

2014

**Catálogo na fonte: Biblioteca da Universidade Federal do
Triângulo Mineiro**

B768i Botta, Gustavo Marino
Implantação do modelo dinâmico inercial rígido em sistemas hidráulicos
no software EPANET / Gustavo Marino Botta. -- 2014.
63 f. : il., graf.

Dissertação (Mestrado Profissional em Inovação Tecnológica). -- Uni-
versidade Federal do Triângulo Mineiro, Uberaba, MG, 2014
Orientador: Prof. Dr. Luiz Fernando Resende dos Santos Anjo

1. Hidráulica. 2. Hidráulica - Simulação por computador. 3. Software - Tes-
tes. 4. Pesquisa experimental. I. Anjo, Luiz Fernando Resende dos Santos. II.
Universidade Federal do Triângulo Mineiro. III. Título.

CDU 626

GUSTAVO MARINO BOTTA

IMPLANTAÇÃO DO MODELO DINÂMICO INERCIAL RÍGIDO EM
SISTEMAS HIDRÁULICOS NO SOFTWARE EPANET

Trabalho de conclusão apresentado ao
Programa de Mestrado Profissional em
Inovação Tecnológica da Universidade
Federal do Triângulo Mineiro, como requisito
para obtenção do título de mestre.

Uberaba, 12 de dezembro de 2014

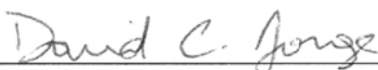
Banca Examinadora:



Prof. Dr. Luiz Fernando Resende dos Santos Anjo
Orientador – PMPIT - UFTM



Prof. Dr. Nélio Muniz Mendes Alves Cruz - IFTM
Membro titular – IFTM



Prof. Dr. David Calhau Jorge
Membro Titular – UFTM

Dedico este trabalho à minha família que sempre me apoiou nas ausências da convivência familiar que foram necessárias para dedicar a este trabalho.

AGRADECIMENTOS

À Universidade Federal do Triângulo Mineiro pela oportunidade de desenvolvimento desta pesquisa.

Ao Prof. Dr. Luiz Fernando Resende dos Santos Anjo, pela sua orientação, dedicação e paciência na construção deste conhecimento.

À coordenação do PMPIT - Programa de Mestrado Profissional em Inovação Tecnológica pela oportunidade de vivenciar a construção de várias formas de trabalhos científicos.

Aos professores do PMPIT, pelos conhecimentos transmitidos.

Ao professor Geovany Araújo Borges por permitir o uso da biblioteca GMATRIX.

Para a minha família pela paciência e apoio fornecidos nesta etapa de minha vida.

"Comece fazendo o que é necessário, depois o que é possível,
e de repente você estará fazendo o impossível".

São Francisco de Assis

RESUMO

Como problema de pesquisa, foi buscado desenvolver a possibilidade de simular eventos hidráulicos na rede de distribuição levando em consideração o efeito da inércia. O software EPANET que tem como vantagens: ser de domínio público, ser utilizado em diversos trabalhos científicos, ser um dos modelos mais utilizados em diversos países, ser o mais confiável e ter compatibilidade com vários outros simuladores. Ainda assim possui a desvantagem de não utilizar em seus cálculos os efeitos da inércia. Como objetivo deste trabalho, teve-se a implementação, no software EPANET, da simulação do Modelo Dinâmico Inercial Rígido (MDIR) para análise de escoamentos transitórios lentos em regime não permanente em instalações a condutos forçados e posteriormente, a comparação do modelo MDIR com o modelo Método do Gradiente (método de cálculo usado pelo EPANET). Como metodologia, foram adquiridos os arquivos fontes do software EPANET, realizado a análise da engenharia de software, elaborado os algoritmos para o modelo MDIR e feito a pesquisa experimental comparando o Modelo MDIR com o Método do Gradiente. Esta simulação é importante, pois o acompanhamento da carga e da vazão nas instalações hidráulicas é necessário para detectar os níveis de abastecimento e os picos de pressão que podem ocasionar rompimentos na tubulação. Com a introdução do MDIR, no software EPANET, esta falha ficou resolvida, pois o MDIR melhorou o motor de cálculo do EPANET utilizando algoritmos que consideram os efeitos da inércia. Desta forma, agora se pode detectar picos de pressão e prever rompimentos das tubulações ou de seus acessórios. A ampliação do Software EPANET foi possível onde o MDIR simula o modelo dinâmico levando em consideração a inércia. A simulação no cotidiano foi aprimorada visto que além de contribuir para soluções rápidas e com custos menores, também pode prever picos de pressão e possíveis rompimentos das tubulações. Desta forma, conclui-se que para o estudo de sistemas hidráulicos o modelo dinâmico inercial rígido (MDIR), se apresenta como uma solução muito interessante.

Palavras-chave: EPANET, MDIR, Simulação hidráulica.

ABSTRACT

As a research problem, we sought to develop the ability to simulate hydraulic events in the distribution network taking into account the effect of inertia. The EPANET software has advantages such as: be in the public domain, to be used in several scientific essays, be one of the models most used in different countries, the most reliable and have compatibility with several other simulators. Still has the disadvantage to do not use in their calculations the effects of inertia. As the objective of this work, we had the implementation of the EPANET software, the simulation of the Dynamic Model Inertial Drive software (DMID) for analysis of slow transient flows in non-steady state facilities to penstocks and subsequently comparing the model DMID with the Gradient Method model (method of calculation used by EPANET). As a methodology, source files were acquired from EPANET software, conducted the analysis of software engineering, developed the algorithms for DMID model and made experimental research comparing the Model DMID with Gradient Method. This simulation is important, as monitoring the load and the flow on hydraulic installation is required to detect the supply levels and pressure surges that can cause disruptions in the pipeline. With the introduction of DMID, in the EPANET software, this fault has been resolved, because the DMID improved EPANET calculation engine using algorithms that consider the effects of inertia. Thus, now it can be detected pressure peaks and prevent disruptions of pipes or fittings. The expansion of the Software EPANET was possible where the DMID simulates the dynamic model taking into account the inertia. The simulation in daily life has been improved seen that besides the contribution to quick solutions with now costs can also provide pressure peaks and possible disruptions of pipes. Thus, it is concluded that for the study of the hydraulic system, the Dynamic Model Inertial Drive software (DMID) is presented as a very interesting solution.

Keywords: EPANET, DMID, hydraulic simulation.

LISTA DE ILUSTRAÇÕES

Figura 1 – EPANET como base para softwares comerciais.....	18
Figura 2 – Componente do algoritmo de cálculo do MDIR.....	18
Figura 3 – Estrutura de migração para modelo MDIR.....	19
Figura 4 – Um exemplo de hashing.....	28
Figura 5 – Hashing usado no EPANET.....	29
Figura 6 - Exemplo de rede de tubulação no EPANET.....	30
Figura 7 – Tela parâmetros do debug no DEV-C++.....	32
Figura 8 – Adaptação na barra de ferramentas.....	32
Figura 9 – Adaptação na barra de menus.....	32
Figura 10 – Depuração: Node.....	35
Figura 11 – Depurando elemento inicial (1ª forma).....	35
Figura 12 – Depurando elemento inicial (2ª forma).....	36
Figura 13 – Depurando elemento específico.....	36
Figura 14 – Depurando uma lista na estrutura (1º elemento).....	37
Figura 15 – Depurando uma lista na estrutura (2º elemento).....	37
Figura 16 – Depurando uma lista na estrutura (3º elemento - inexistente).....	37
Figura 17 – Depurando elemento do vetor específico.....	38
Figura 18 – Depurando o vetor completo.....	38
Figura 19 – Estrutura GMATRIX.....	41
Figura 20 – Visualizando os dados da matriz.....	41
Figura 21 – Leiaute da rede 01.....	42
Figura 22 – Configuração dos tempos de cálculos da rede 01.....	43
Figura 23 – Vazão do tubo 3 - Gradiente.....	43
Figura 24 – Vazão do tubo 3 - MDIR.....	44
Figura 25 – Leiaute da rede 02.....	44
Figura 26 – Configuração dos tempos de cálculos rede 02.....	45
Figura 27 – Configuração do padrão de tempo da rede 02.....	46
Figura 28 – Carga hidráulica do nó 2 - Gradiente.....	46
Figura 29 – Carga hidráulica do nó 2 - MDIR.....	47
Figura 30 – Vazão do tubo 1 - Gradiente.....	47
Figura 31 – Vazão do tubo 1 - MDIR.....	48

Figura 32 – Retirando tempo inicial do gráfico da rede 02	48
Figura 33 – Vazão do tubo 1 – MDIR – cortando instantes iniciais	49
Figura 34 – Leiaute da rede 03	49
Figura 35 – Configuração dos tempos de cálculos da rede 03	50
Figura 36 – Carga hidráulica do tanque 1 – Gradiente.....	50
Figura 37 – Carga hidráulica do tanque 1 – MDIR	51
Figura 38 – Vazão do tubo 1 – Gradiente.....	51
Figura 39 – Vazão do tubo 1 – MDIR.....	52
Figura 40 – Configuração dos tempos de cálculos da rede 04	53
Figura 41 – Carga hidráulica do tanque 1 – Gradiente.....	53
Figura 42 – Carga hidráulica do tanque 1 – MDIR	54
Figura 43 – Vazão do tubo 1 – Gradiente.....	54
Figura 44 – Vazão do tubo 1 – MDIR	55
Figura 45 – Leiaute da rede 05	56
Figura 46 – Configuração dos tempos de cálculos da rede 05	56
Figura 47 – Carga hidráulica do tanque 7 – Gradiente.....	57
Figura 48 – Carga hidráulica do tanque 7 – MDIR	57
Figura 49 – Vazão do tubo 2 – Gradiente.....	57
Figura 50 – Vazão do tubo 2 – MDIR.....	58

LISTA DE QUADROS

Quadro 1: Limite de escoamento utilizado pelo EPANET.....	27
Quadro 2: Principais alterações na rotina RunHydraulics.....	33
Quadro 3: Declaração de funções externas na seção interface (epanet2.pas)	34
Quadro 4: Declaração de funções externas na seção implementation (epanet2.pas)	34
Quadro 5: Exemplo de adaptação no arquivo epanet.c	39
Quadro 6: Ponteiros de matrizes criadas em vars.h.....	39
Quadro 7: Declaração de ponteiros do tipo PGMATRIX	40
Quadro 8: Alocação dos ponteiros do tipo PGMATRIX	40

LISTA DE SIGLAS

DEV-C++ – Ambiente gráfico de desenvolvimento para linguagem C/C++.

DLL – Dynamic Link Libraries – Biblioteca de Vinculação Dinâmica

EPANET – Software de simulação de redes de distribuição hidráulica

LARA – Laboratório de Automação e Robótica

mca – metros de coluna d'água

MDIR – Modelo Dinâmico Inercial Rígido

NBR - denominação de norma da Associação Brasileira de Normas Técnicas (ABNT)

UnB – Universidade de Brasília

SUMÁRIO

1 INTRODUÇÃO.....	13
1.1 OBJETIVOS.....	15
2 CONTRIBUIÇÃO DESTE TRABALHO PARA A SIMULAÇÃO HIDRÁULICA ..	17
3 REVISÃO BIBLIOGRÁFICA	20
3.1 MODELO DINÂMICO INERCIAL RÍGIDO.....	20
3.2 ENGENHARIA REVERSA.....	26
3.3 TIPOS DE ESCOAMENTOS	26
3.4 TABELA HASH	27
4 DESENVOLVIMENTO DO MDIR.....	30
4.1 ENGENHARIA REVERSA – DEPURAÇÃO	30
4.2 ADAPTAÇÃO NO AMBIENTE DELPHI.....	32
4.3 TÉCNICAS DE DEPURAÇÃO NO DEV-C++	34
4.4 ADAPTAÇÃO NA DLL.....	38
4.5 GMATRIX	39
5 RESULTADOS COMPARATIVO DO MDIR COM O METÓDO GRADIENTE....	42
5.1 SIMULAÇÃO REDE 01 SEM DEMANDA	42
5.2 SIMULAÇÃO REDE 02.....	44
5.3 SIMULAÇÃO REDE 03.....	49
5.4 SIMULAÇÃO REDE 04.....	52
5.5 SIMULAÇÃO REDE 05.....	55
6 CONCLUSÃO.....	59
REFERÊNCIAS	61
ANEXO A – AUTORIZAÇÃO PARA USO DO GMATRIX	63

1 INTRODUÇÃO

Segundo Tsutiya (2006) o abastecimento de água é dividido nas seguintes partes: manancial, captação, estação elevatória, adutora, estação de tratamento de água, reservatório e rede de distribuição.

O manancial é a origem de onde é retirada a água para o abastecimento como, por exemplo, os rios.

A captação são os equipamentos instalados junto ao manancial para extrair a água de forma adequada. A captação é a primeira unidade do sistema de abastecimento de água e seu funcionamento é importante, pois dele é que se desencadeia o restante do processo.

A estação elevatória é o conjunto de obras e equipamentos para deslocar a água para a unidade seguinte. São unidades providas de bombas hidráulicas e tanques que elevam e aumentam a pressão da água.

A adutora é a canalização que conduz a água entre as unidades, isto é, o encanamento destinado a conduzir as águas.

A estação de tratamento de água é o conjunto destinado a tratar a água para o consumo. Nesta etapa é feito um tratamento químico, a floculação, a decantação e a filtração. Segundo a (SABESP), o conjunto destas 3 últimas etapas é chamado de clarificação, pois nelas são retiradas as partículas de impurezas deixando a água límpida.

O reservatório é a parte do sistema de distribuição de água que regula as condições de pressões na rede de distribuição. Inicialmente a água é armazenada em reservatórios de distribuição e posteriormente em reservatórios de bairros, espalhados em pontos estratégicos na cidade.

Conforme Tsutiya (2006), a rede de distribuição são as tubulações e acessórios que disponibilizam água aos consumidores. É a estrutura do sistema mais próxima à realidade urbana, e a mais dispendiosa compreendendo cerca de 50 a 75% do custo total. É constituída de um conjunto de tubulações interligadas instaladas ao longo das vias públicas, junto aos edifícios, conduzindo a água aos pontos de consumo (moradias, escolas, hospitais).

Este projeto propõe a implementação de uma simulação hidráulica utilizando o Modelo Dinâmico Inercial Rígido que abrange as duas últimas etapas que são: o reservatório e a rede de distribuição. Será considerada a água já tratada e será realizada a simulação com a pressão do reservatório na rede de distribuição.

Esta simulação é importante, pois o acompanhamento da carga e da vazão no decorrer do tempo nas instalações hidráulicas é necessária quando se estuda os resultados de manobras que alteram as condições de escoamento (Anjo, 2008).

A análise do escoamento de fluido através de um sistema hidráulico a condutos forçados é usualmente feita de forma unidimensional segundo o eixo dos condutos, seguindo diferentes hipóteses simplificadoras, que permitem a concepção de distintos modelos. Estes modelos são classificados, como observa Cabrera, et al. (1994), em modelos dinâmicos e estáticos, segundo a variação ou não do escoamento no tempo. Os modelos dinâmicos se subdividem em modelos dinâmicos inerciais e não inerciais, caso levem ou não a inércia como elemento na variação temporal. Os modelos dinâmicos inerciais podem, por sua vez, considerar ou não efeitos de deformação do fluido e do conduto como decorrência, o que leva a sub-classificá-los em elásticos ou não elásticos (rígidos). (ANJO, 2008)

Conforme exemplificado na citação anterior, o modelo dinâmico inercial elástico seria o mais fidedigno nos resultados, pois este modelo considera a variação de velocidade do fluido, o efeito da inércia e também a elasticidade e a compressibilidade da água e da tubulação. No entanto, como o modelo rígido não prejudica a análise final conforme pode ser observado nas citações a seguir, o mesmo foi escolhido para utilização na simulação hidráulica deste projeto.

Analisando a fórmula do módulo volumétrico $B = -\frac{\Delta P}{\Delta V/V}$, onde:

ΔP = diferença de pressão.

ΔV = diferença de volume.

V = volume.

Quanto mais difícil for comprimir um material, menor será o decréscimo relativo de volume. A compressibilidade é o inverso do módulo volumétrico. Como os líquidos são relativamente incompressíveis, eles possuem grandes valores de B que são independentes da temperatura e da pressão (TIPLER e MOSCA, 2012, p. 434).

Brunetti (2008, p. 10) também defende a utilização de fluido incompressível:

“Diz-se que um fluido é incompressível se o seu volume não varia ao modificar a pressão. Isso implica o fato de que, se o fluido for incompressível, a sua massa específica não variará com a pressão.

É claro que na prática não existem fluidos nessas condições. Os líquidos, porém, têm um comportamento muito próximo a esse e na prática, normalmente, são considerados como tais. [...]

[...] Sempre que ao longo do escoamento a variação da massa específica ρ for desprezível, o estudo do fluido será efetuado pelas leis estabelecidas para fluidos incompressíveis.”

O EPANET é um software que simula sistemas de tubulação de distribuição de água. É um software de domínio público que pode ser livremente copiado e distribuído. É um programa desenvolvido para Windows. O EPANET realiza simulação ao longo do período de movimento da água dentro de redes de tubulação pressurizada. (EPA - United States Environmental Protection Agency, 2013, tradução nossa)

A simulação é importante para a prática cotidiana. Heller e Pádua (2010, p. 756) enfatizam isto destacando que os problemas operacionais devem ser resolvidos com mais rapidez e por isto não podem depender de licitações. Várias vezes são resolvidos com equipamentos e tubulações existentes no estoque, então em vez de fazer um projeto se faz uma simulação com os recursos existentes. Os modelos de simulação contribuem desta forma para soluções rápidas e com custos menores.

O problema abordado neste trabalho é desenvolver a possibilidade de simular eventos hidráulicos na rede de distribuição levando em consideração o efeito da inércia.

1.1 OBJETIVOS

Como objetivo, é implementado, no software EPANET, a simulação do Modelo Dinâmico Inercial Rígido (MDIR) - modelo elaborado por Anjo (2008) - para análise de escoamentos transitórios lentos em regime não permanente em instalações a condutos forçados e posteriormente, comparado o modelo MDIR com o modelo Método do Gradiente. Conforme Rossman (2000), o software EPANET utiliza como método de cálculo o método gradiente.

Conforme Netto et al. (1998), o processamento de dados com o auxílio de computadores, tem contribuído no projeto e implantação de obras hidráulicas e propiciando a modelagem de simulações que permitem prever e analisar fenômenos dinâmicos.

A metodologia utilizada foi a aquisição dos arquivos fontes do software EPANET visto que os mesmos são de domínio público. A análise da engenharia de software (programação) utilizada no sistema original EPANET para determinar a forma de armazenamentos dos dados das tubulações, reservatórios, bombas e outros. Elaboração dos algoritmos responsáveis pela codificação das fórmulas do modelo proposto (Modelo Dinâmico Inercial Rígido - MDIR). Realização de uma pesquisa experimental onde foram validados os resultados com uma série de simulações cujos valores foram comparados o

Modelo MDIR com o modelo padrão utilizados no software EPANET (Método do Gradiente).

2 CONTRIBUIÇÃO DESTE TRABALHO PARA A SIMULAÇÃO HIDRÁULICA

A área de engenharia tem se utilizado de ferramentas computacionais para auxiliar o alcance dos objetivos almejados. Com a utilização do processamento de dados, os engenheiros conseguem obter resultados de forma automatizada de tarefas rotineiras obtendo assim uma economia de tempo considerável. Como a simulação gera resultados rápidos os mesmos podem ser comparados com outras simulações proporcionando assim o encontro de soluções mais apropriadas e melhores.

Os programas de simulação hidráulica aumentaram em quantidade devido às vantagens destes tipos de softwares. Olaia (2012), realizou um estudo sobre diversos softwares de simulação dentre eles: o Hidrocad, Epanet, Watercad, Strumap, Synergiee Water e Inssaa. A mesma destaca a grande aceitação do EPANET:

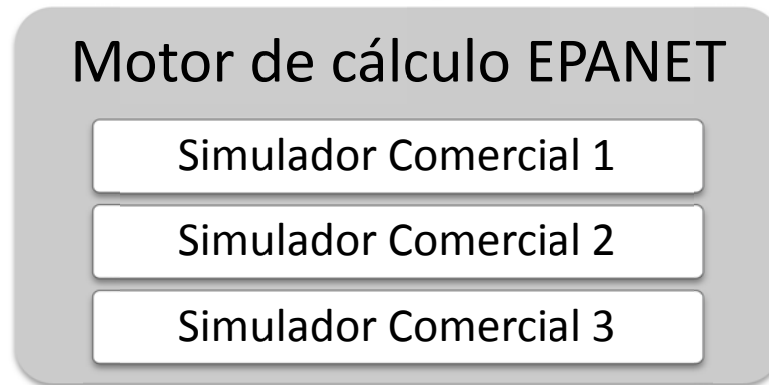
[...] Na medida em que este é um programa de domínio público, o mesmo tem sido muito aprofundado em trabalhos científicos, sendo um dos modelos mais utilizados para simulação hidráulica em muitos países (PINTO, 2009), nomeadamente em Portugal. (OLAIA, 2012, p. 15)

Acrescenta também que “Este é atualmente, um dos mais fiáveis programas para este tipo de estudos [...]” (OLAIA, 2012, p. 15) e por fim elege o EPANET como o software que será utilizado para as simulações de seus estudos de mestrado.

Além do software EPANET ser um destaque no cenário mundial, o mesmo possui compatibilidade com outros simuladores.

Segundo Heller e Pádua (2010, p. 753) os algoritmos de cálculo tornaram, definitivamente, a ferramenta de trabalho dos engenheiros responsáveis pelos sistemas de abastecimento de água das maiores e melhores empresas do mundo. E que o EPANET é a base para grande parte dos softwares comerciais. Sendo assim o EPANET pode ser considerado como o motor de cálculo para outros programas de simulação (Figura 1) e cujos custos de aquisições são elevados se observado como exemplo o Watercad (OLAIA, 2012).

Figura 1 – EPANET como base para softwares comerciais

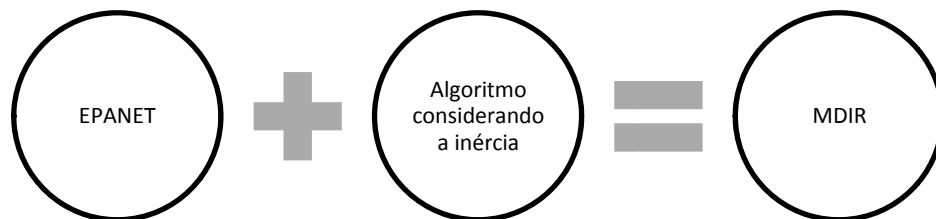


Fonte: Do autor, 2014

Conforme Gomes (2010, p. 214) o EPANET apresenta limitações nos resultados de seus experimentos devido ao fato do modelo hidráulico (método gradiente) não considerar os efeitos de inércia em um sistema hidráulico.

O algoritmo de cálculos MDIR deste trabalho (veja Figura 2) efetua a melhora neste motor de cálculo do EPANET, pois passa a considerar o efeito da inércia nos cálculos.

Figura 2 – Componente do algoritmo de cálculo do MDIR



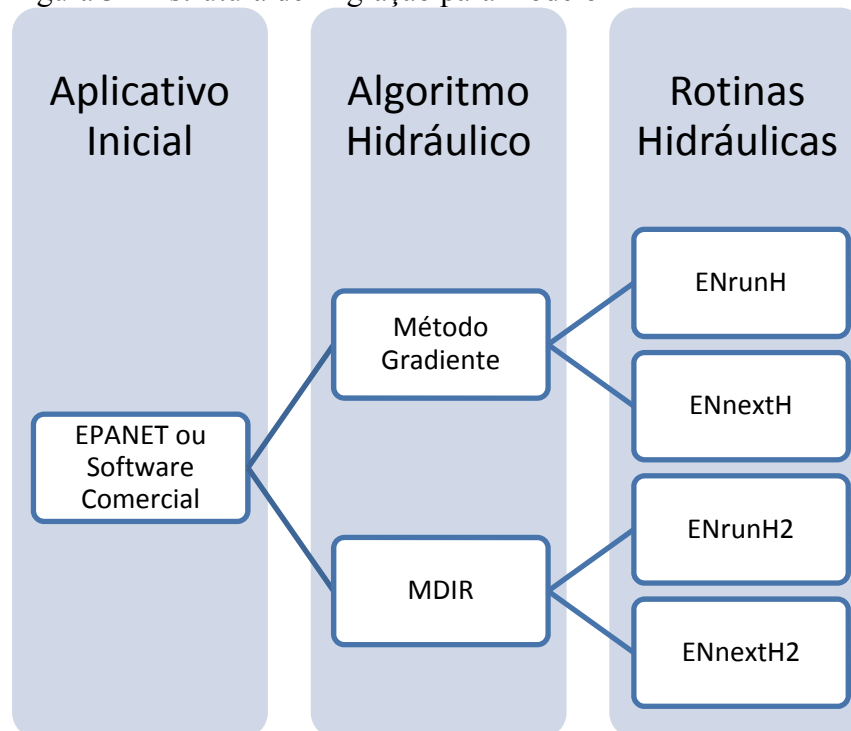
Fonte: Do autor, 2014

Vale a pena ressaltar a importância de controlar a pressão na rede de distribuição de água, que, segundo a NBR 12.218/1994 (projeto de rede de distribuição de água para abastecimento público), deve-se ter valor não superior a 10 mca para pressão dinâmica nos pontos das tubulações distribuidoras. Assim, é indispensável o acompanhamento do desempenho hidráulico da rede de distribuição, pois isso permite prever as alterações de vazão e de pressão da água (GOMES, 2010, p. 124). O modelo MDIR auxilia neste acompanhamento onde é vital detectar picos de pressão que possam ocasionar rompimentos das tubulações ou de seus acessórios.

Como o uso do MDIR busca aperfeiçoar os resultados obtidos com o EPANET, desde o início do projeto teve-se a preocupação de manter a compatibilidade das funções mantendo o método gradiente e acrescentando o MDIR, pois assim o pesquisador poderá rodar os dois métodos para efetuar as comparações (como foi feito neste trabalho).

Visualizando a possibilidade do algoritmo MDIR ser interessante para utilização em redes, as funções de chamadas do núcleo de cálculo do EPANET somente foram acrescentadas com o sufixo 2, por exemplo: a função ENrunH passou a se chamar ENrunH2. Isto facilitará a atualização dos softwares comerciais que utilizam o EPANET como base de cálculo (veja Figura 3), pois bastará utilizar o recurso localizar/substituir que rapidamente se consegue alternar do método gradiente para o método MDIR quando o mesmo estiver em um desenvolvimento completo e bem assimilado pela comunidade acadêmica.

Figura 3 – Estrutura de migração para modelo MDIR



Fonte: Do autor, 2014

3 REVISÃO BIBLIOGRÁFICA

No tópico 3.1, a equação da continuidade que será apresentada segue a afirmação de (BRUNETTI, 2008, p. 75): “[...] ao longo do escoamento, velocidades médias e áreas são inversamente proporcionais, isto é, à diminuição da área correspondem aumentos da velocidade média na seção e vice-versa.”, mas em vez de velocidade média entenda velocidade instantânea.

Também será utilizada a equação da quantidade de movimento que conforme Brunetti (2008, p. 121):

Essa equação nada mais é que a segunda lei de Newton da dinâmica modificada funcionalmente para o estudo da Mecânica dos Fluidos. Segundo essa lei, a aceleração de uma certa massa implica a existência de uma força resultante sobre ela que tem, em cada instante, a direção e o sentido da aceleração. Acelerar uma massa significa modificar sua velocidade em módulo e/ou direção, e por essa observação, para que a velocidade de um fluido seja modificada em módulo ou direção, será necessário aplicar uma força provocada por algum agente externo, em geral uma superfície sólida em contato com o escoamento. Pelo princípio da ação e reação, se a superfície aplica uma força no fluido, este aplicará, sobre a superfície, uma outra de mesmo módulo e de sentido contrário. A observação desses fatos permitirá a construção da equação da quantidade de movimento, nos moldes desejados.

3.1 MODELO DINÂMICO INERCIAL RÍGIDO

Este subtítulo é todo baseado na tese de Anjo (2008), sob o título Modelo Hidráulico para Transitórios Lentos em Conduto Forçado, onde o mesmo descreve o MDIR. O referido tópico contém algumas adaptações onde foram colocadas as comparações ao modelo gradiente diretamente no modelo MDIR.

As equações gerais que governam o escoamento fluido nos condutos de uma instalação hidráulica a pressão, permitem a determinação das variáveis de estado; carga e vazão ao longo da tubulação no transcorrer do tempo; $H = H(x, t)$ e $Q = Q(x, t)$, onde x é a distância medida ao longo do eixo da tubulação e t o tempo transcorrido. Sob as hipóteses do modelo dinâmico inercial elástico, estas equações podem ser expressas através das equações da continuidade e da quantidade de movimento na forma (ANJO, 2008 apud STREETER, 1993):

$$\left\{ \begin{array}{l} \frac{gA}{a^2} \frac{\partial H}{\partial t} + \frac{\partial Q}{\partial x} = 0 \quad \text{continuidade} \\ \frac{\partial H}{\partial x} + \frac{1}{gA} \frac{\partial Q}{\partial t} + f \frac{Q|Q|}{2gDA^2} = 0 \quad \text{quantidade de movimento} \end{array} \right. \quad (1a,b)$$

onde:

g = aceleração da gravidade.

A = área da seção transversal do tubo.

a = celeridade.

H = carga piezométrica no nó.

t = tempo transcorrido.

Q = descarga em volume nos tubos.

x = distância medida ao longo do eixo da tubulação.

f = fator de atrito da fórmula universal de perda de carga.

D = diâmetro do tubo.

Sob a hipótese de modelo rígido, a celeridade de propagação (a) torna-se infinita, levando a zero o primeiro termo do lado esquerdo da equação da continuidade, reduzindo esta equação a $\partial Q / \partial x = 0$, de onde conclui-se que a vazão será a mesma em toda a extensão da tubulação para cada instante de tempo t , logo $Q = Q(t)$ e $\partial Q / \partial t = dQ / dt$.

A equação da quantidade de movimento integrada entre as seções extremas da tubulação, (1) seção de montante e (2) seção de jusante, distantes um comprimento L medido ao longo de seu eixo, permite obter:

$$(H_2 - H_1) + f \frac{LQ|Q|}{2gDA^2} = - \frac{L}{gA} \frac{dQ}{dt} \quad (2)$$

resultando na equação que governa o modelo dinâmico inercial rígido.

A equação do modelo rígido pode ser expressa na seguinte forma matricial:

$$A_{12}H + A_{10}H_0 + f(Q) = -\beta \frac{dQ}{dt} \quad (3)$$

onde:

$A_{12} = A_{21}^T$ ($np \times nn$) - matriz de incidência de incógnitas cargas nodais.

$H^T = \{H_1, \dots, H_p\}$ ($1 \times nn$) – cargas nodais incógnitas.

A_{10} ($np \times no$) = matriz de incidência de nós com carga fixa.

$H_0^T = \{H_0^1, \dots, H_0^{no}\}$ ($1 \times no$) – nós com cargas estabelecidas (conhecidas ou fixas)

$f(Q)^T = \{f(Q_1), \dots, f(Q_{np})\}$ – lei que expressa a variação de carga nos tubos.

β é uma matriz diagonal quadrada de ordem np , com o elemento (i, i) , formado pela relação obtida do tubo i , dada por $\beta_{ii} = L_i / (gA_i)$.

sendo:

nn = número de nós com carga incógnita.

no = número de nós com carga fixa.

np = número de tubos com incógnita vazão.

Os elementos da matriz A_{12} são escritos na forma:

$$A_{12}(i, j) = \begin{cases} 1 & \text{se a vazão no tubo } i \text{ chega ao nó } j \\ 0 & \text{se o tubo } i \text{ e o nó } j \text{ não estão conectados} \\ -1 & \text{se a vazão no tubo } i \text{ sai do nó } j \end{cases} \quad (4)$$

e os elementos de A_{10} definidos de forma similar a A_{12} , mas em relação aos nós com carga fixa.

Utilizando uma aproximação simples para dQ/dt , na forma: $(Q - Q_0)/\Delta t$, com que Q_0 avaliada em K relativa ao instante (t) e Q avaliada em $(K + 1)$ no instante $(t + \Delta t)$, e considerando que na evolução no tempo o termo de atrito possa ser avaliado em relação ao instante t , pode-se reescrever a equação anterior como:

$$A_{12}H + A_{10}H_0 + GQ_0 + BQ = 0 \quad (5)$$

onde G e B são matrizes diagonais quadradas de ordem np :

$$G = \begin{bmatrix} R_1|Q_{01}| - B_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & R_{np}|Q_{0np}| - B_{np} \end{bmatrix} \quad (6)$$

$$B = \begin{bmatrix} \frac{L_1}{gA_1\Delta t} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{L_{np}}{gA_{np}\Delta t} \end{bmatrix} = \begin{bmatrix} \frac{\beta_1}{\Delta t} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{\beta_{np}}{\Delta t} \end{bmatrix} \quad (7)$$

O sistema de equações a ser resolvido para a solução do problema será:

$$\begin{cases} A_{12}H + BQ = GQ_0 - A_{10}H_0 \\ A_{21}Q = q \end{cases} \quad (8)$$

onde na parte GQ_0 foi trocado o sinal de (-) pelo sinal (+) pois a inversão do sinal foi feita no cálculo interno de G sendo o mesmo calculado por $G=B-R/Q/$ em vez do formato original que seria $G= R/Q/-B$. Assim resultando no sistema de equações:

$$\begin{bmatrix} B & A_{12} \\ A_{21} & 0 \end{bmatrix} \cdot \begin{bmatrix} Q \\ H \end{bmatrix} = \begin{bmatrix} GQ_0 - A_{10}H_0 \\ q \end{bmatrix} \quad (9)$$

que é linear e tendo como uma propriedade importante o fato da matriz dos coeficientes ser constante, ou seja será montada apenas uma vez durante o processo de solução. O vetor de soluções é variável no tempo e dependerá da equação de resistência utilizada. Considerando a Fórmula Universal de perda de carga pode-se escrever:

$$A_{21}B^{-1}GQ_0 - A_{21}B^{-1}A_{12}H - A_{21}B^{-1}A_{10}H_0 = q \quad (10)$$

Separando os termos da equação acima tem-se:

$$A_{21}B^{-1}A_{12}H = A_{21}B^{-1}GQ_0 - A_{21}B^{-1}A_{10}H_0 - q \quad (11)$$

Chamando o produto de matrizes, que multiplica as cargas incógnitas, de J e o vetor resultante do segundo membro de F :

$$JH = F \quad (12)$$

onde a matriz J pode ser escrita como:

$$J_{ij} = \begin{cases} i = j \Rightarrow J_{ij} = \sum_{K \in i} \frac{1}{B_{iK}} \\ i \neq j \Rightarrow J_{ij} = -\frac{1}{B_{ij}} \end{cases} \quad (13)$$

onde J é a matriz onde os elementos da diagonal são:

$$J_{ii} = \sum_i p_{ij} \quad (14)$$

enquanto que os elementos não nulos fora da diagonal são:

$$J_{ij} = -p_{ij} \quad (15)$$

onde p_{ij} é o inverso da derivada de B (7) no trecho entre os nós i e j em relação a vazão:

$$p_{ij} = n_v \frac{1}{B} \quad (16)$$

onde n_v é o expoente da vazão.

Uma vez obtida a solução para o sistema de equação dada por (12), para a determinação do vetor de cargas, pode-se obter as vazões pela equação:

$$Q = B^{-1}GQ_0 - B^{-1}A_{12}H - B^{-1}A_{10}H_0 \quad (17)$$

Resolve-se o sistema de equações para a determinação das cargas, equação (12), com os valores de carga obtidos determinam-se as vazões nos tubos por operações matriciais simples obtidas pela equação (17).

Deve se observar que a matriz inversa que permite a solução do sistema de equações em (12) se mantém constante durante todo o processo, sendo, portanto necessário calculá-la apenas uma vez, pois não está expresso em termos das vazões. Esta observação pode ser importante, pois o fato de não necessitar o recálculo da matriz inversa, se o sistema for demasiadamente grande pode reduzir significativamente o tempo final de processamento.

Quando alguns nós de um sistema hidráulico possuem reservatórios de nível variado, os contornos dos mesmos são realizados fora da equação (12), da seguinte forma:

$$H^{t+\Delta t} = \frac{\sum Q}{A_r} \Delta t + H^t \quad (18)$$

onde A_r é a área da seção transversal do reservatório.

Se o interesse é a de obtenção da condição estática, como faz o modelo estático empregado no EPANET, o modelo proposto também poderá ser empregado. Nesse caso pode se abrir mão da precisão numérica durante a condição transitória para se obter a aceleração da convergência para o regime permanente adotando-se valores de dt maiores. Cabe observar que o dt é a discretização numérica de uma equação diferencial e, portanto para garantir sua adequação ao fenômeno físico este valor deve ser pequeno para garantir a representação da derivada. O uso de dt 's grandes não traduz corretamente a evolução do fenômeno físico, embora para os testes efetuados tenha-se obtido a convergência para o regime permanente final, de forma acelerada.

3.2 ENGENHARIA REVERSA

Como o EPANET é um software de domínio público, os códigos fontes são disponibilizados para estudos e alterações. Para entender o funcionamento do algoritmo utilizado no EPANET, utilizou-se a engenharia reversa.

A engenharia reversa adotada neste trabalho seguiu o seguinte conceito:

“Engenharia Reversa: O processo de análise de um sistema sujeito a: (1) identificar os componentes do sistema e suas inter-relações, e (2) criar representações do sistema em outra forma ou em um nível mais alto de abstração.” (AVERSANO et al., 2005 apud KHOSROW-POUR, 2006, p. 583)

No EPANET, parte do código fonte é desenvolvida em linguagem Delphi e é utilizada uma biblioteca externa, no formato de vinculação dinâmica (Dynamic Link Libraries – DLL). Os cálculos hidráulicos e de qualidade da água são todos feitos nesta biblioteca. Esta biblioteca cujo nome é epanet2.dll foi escrita na linguagem C.

Segundo a Microsoft Developer Network (2014a) existem duas formas de vincular bibliotecas: a vinculação estática e a vinculação dinâmica. Na vinculação estática, as bibliotecas que serão utilizadas no programa são compiladas e anexadas ao código executável deixando o mesmo com tamanho maior e não permitindo que outros programas acessem esta biblioteca que foi embutida no executável. Na vinculação dinâmica a biblioteca fica armazenada externamente ao arquivo executável principal como, por exemplo, em um arquivo com extensão DLL. No módulo executável é necessário somente ter os dados que consigam localizar esta função externa não sendo necessário a mesma estar embutida no código executável. Esta localização será feita somente em tempo de execução (*run-time*) do programa e caso não encontre o arquivo referenciado é neste momento que será acusado o erro. A vinculação dinâmica fornece uma maneira para um aplicativo chamar uma função que não é parte do seu código executável. O código executável para a função está localizado em uma DLL, que contém um ou mais funções que são compiladas, vinculadas e armazenadas separadamente dos processos que irão usá-las.

3.3 TIPOS DE ESCOAMENTOS

Segundo Fialho (2003, p. 80) existem 3 tipos de escoamento: laminar, turbulento e indeterminado. O escoamento laminar mantém linhas de fluxos uniformes devido ao número de Reynolds menor que está relacionado à mais influência da viscosidade do fluido. O

escoamento turbulento possui linhas de fluxos desordenados devido ao número de Reynolds elevado e que caracteriza atuação da força de inércia e maior perda de carga. E por último o escoamento indeterminado ou limite crítico do escoamento é quando o número de Reynolds é médio (nem alto, nem baixo) sendo impossível caracterizar como escoamento totalmente laminar ou turbulento. Porto (2006, p. 3) afirma que problemas que usam a água como líquido são geralmente classificados como turbulentos devido à sua viscosidade baixa.

O software EPANET já calcula este 3 tipos de escoamentos onde utiliza os seguintes números de Reynolds para definir os cálculos para cada tipo de escoamento, veja Quadro 1.

Quadro 1: Limite de escoamento utilizado pelo EPANET

Limites de Escoamento	
Escoamento Laminar	$Re \leq 2000$
Escoamento Indeterminada	$2000 < Re < 4000$
Escoamento Turbulento	$Re \geq 4000$

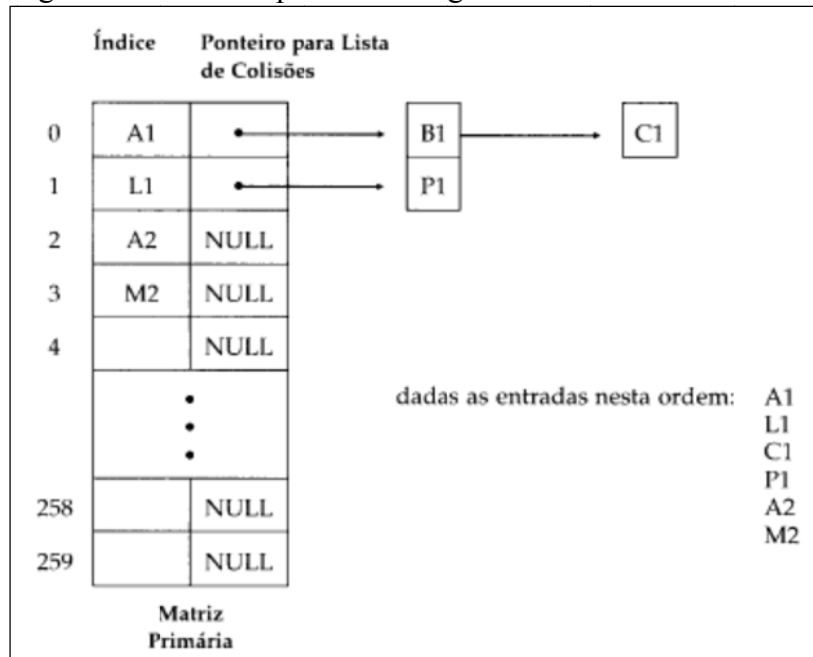
Fonte: Do Autor, 2014

3.4 TABELA HASH

No EPANET são utilizadas duas variáveis para tabela hash: Nht e Lht. A Nht é o acrônimo de Node Hash Table (tabela hash para nós) e o Lht é o acrônimo de Link Hash Table (tabela hash para ligações). Ambas utilizam a mesma forma de armazenamento.

Hashing é o processo de extrair um índice para definir o local de armazenamento de um elemento na matriz. O índice gerado é chamado hash. É utilizada uma matriz física menor, também chamada de matriz primária. Se a posição referenciada pelo índice está livre, o valor é armazenado. Se a posição referenciada pelo índice já está ocupado, ocorre o que se chama colisão, neste caso cria-se uma lista encadeada a partir desta posição para armazenar o valor colidido. Esta lista também pode ser chamada de lista de colisão, veja Figura 4 (SCHILDT, 1997).

Figura 4 – Um exemplo de hashing

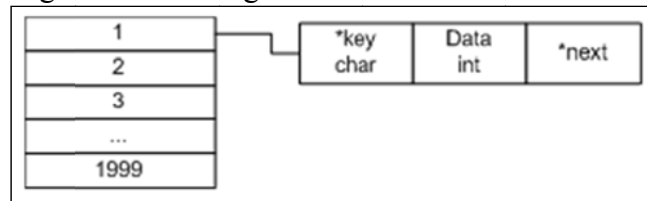


Fonte: Schildt, 1997

No melhor caso (muito raro), somente a matriz primária é utilizada e o acesso se torna rápido. No pior caso (também raro), é usado somente um elemento da matriz primária e o restante são alocados na lista de colisão, o acesso neste caso se torna lento. No caso médio (e o mais provável) o acesso é pelo tempo médio. É importante assegurar que o algoritmo distribua os índices físicos uniformemente (SCHILDT, 1997).

Como citado no início do item, as tabelas hash Nht e Lht utilizam a mesma forma de armazenamento. O EPANET define a matriz primária com 1999 elementos e em caso de colisão expande numa lista encadeada. Cada elemento, na matriz principal e na lista de colisão, armazena os campos Key, Data e Next. O campo Next serve para fazer a ligação da lista encadeada. Os campos Key e Data armazenam respectivamente o nome do elemento (que exibido na tela do EPANET) e o índice que é utilizado para fazer os cálculos. Antes de fazer os cálculos esta tabela é alimentada e após a finalização dos cálculos ela é consultada para indicar os valores corretos a cada elemento nomeado.

Figura 5 – Hashing usado no EPANET



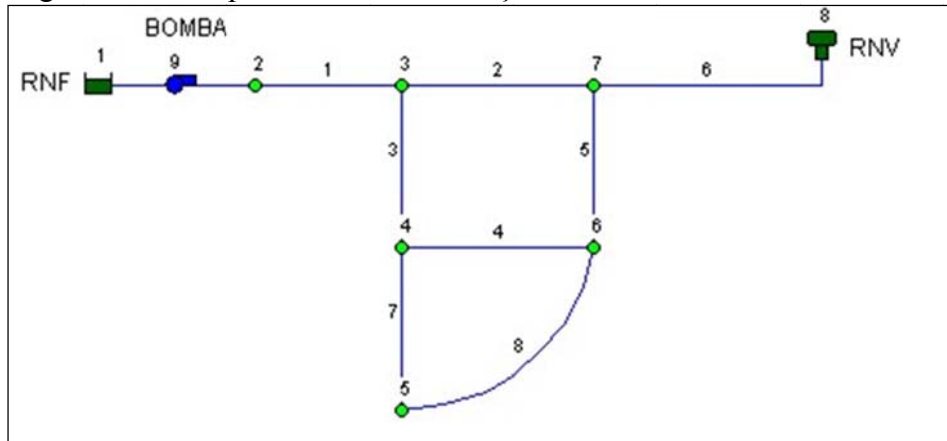
Fonte: Do autor, 2014

Uma técnica que foi utilizada para facilitar a depuração do programa foi nomear os elementos com os mesmos números do campo Data. Isto facilitou o acompanhamento dos cálculos visto que os nomes internos (índices – key) coincidiam com as legendas na rede de distribuição desenhada.

4 DESENVOLVIMENTO DO MDIR

A rede de tubulação é composta de tubos, nós (junções), bombas, válvulas e tanques de armazenamento ou reservatórios (Figura 6). O EPANET fornece: o fluxo de água em cada tubo, a pressão em cada nó e a altura de água em cada tanque.

Figura 6 - Exemplo de rede de tubulação no EPANET



Fonte: Software EPANET, 2013

Conforme Heller e Pádua (2010, p. 760), o EPANET é base para quase todos os *softwares* de simulação de sistemas de abastecimento de água em regime de conduto forçado e o mesmo possui como facilitador o recurso de importar e exportar arquivos do tipo “.INP”. O arquivo “.INP” é um arquivo texto que armazena todos os elementos gráficos e não gráficos, como: coordenadas dos nós, diâmetros dos tubos e níveis dos reservatórios.

4.1 ENGENHARIA REVERSA – DEPURAÇÃO

Para realizar o estudo de engenharia reversa inicialmente se buscou fazer a depuração do programa, visto que este recurso auxilia o programador a entender o código fonte, pois com ele é possível ir passo a passo pela execução do programa visualizando a etapa em que se encontra no código fonte. A depuração também será importante quando for feita a implementação do modelo MDIR e conforme Warner e Goldsman (1996, p. 77) “Uma grande parte do tempo gasto em programas de computador é gasto em depuração.”

Foi obtido o código fonte do EPANET no site da EPA - United States Environmental Protection Agency (2013), na seção downloads. O item *EPANET 2 source code files* contém o

código fonte em Delphi e seus respectivos componentes adicionais e possui também o código fonte em C da DLL: epanet2.dll.

A depuração no Delphi foi executada com sucesso, porém não depurava a DLL. Isto aconteceu porque a DLL estava em outra linguagem.

No site da Epanet.de Hydraulic Network Analysis (MACKE, 2014) foi possível baixar o projeto para construir a DLL do EPANET na ferramenta de desenvolvimento DEV-C++, pois o código fonte foi disponibilizado, mas a junção dos mesmos em DLL estava inicialmente documentada somente para outros compiladores que não eram de uso e conhecimento do autor.

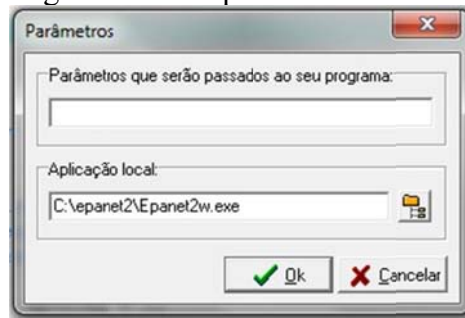
Para conseguir depurar a DLL seguiu-se as instruções da Microsoft Developer Network (2014b), onde indica que a depuração da DLL pode ser obtida iniciando a depuração do projeto que cria o executável que chama a DLL ou o projeto que cria a própria DLL. E a informação de Leão (2003, p. 652)

Uma DLL pode ser compilada da mesma maneira que um projeto de aplicativo e, após compilada, poderá ser livremente distribuída. Você não pode, no entanto, executar uma DLL a partir do ambiente de desenvolvimento do Delphi, selecionando o item Run do menu Run (embora esse item esteja habilitado), pois uma DLL não é uma aplicação (a menos que você defina um aplicativo como Host, a ser especificado na caixa de diálogo Run Parameters, exibida quando se seleciona o item Parameters do menu Run).

Utilizando esta linha de raciocínio de Leão (2003), foi feito um experimento abrindo o código fonte da DLL, em linguagem C, na IDE DEV-C++ e tentado localizar um executável chamador. Foi obtido sucesso, onde o executável principal rodava e quando passava pelas rotinas da DLL a mesma era depurada.

Para conseguir este resultado foi utilizada a seguinte configuração do DEV-C++: no menu *Debug*, no item Parâmetros, no campo aplicação local foi especificado o localização do arquivo executável principal, veja Figura 7. Desta forma a aplicação principal foi executada sem depuração e quando chamava a DLL, a mesma era depurada através da utilização do comando menu *Debug*, item *Debug* ou a tecla de atalho F8.

Figura 7 – Tela parâmetros do debug no DEV-C++.



Fonte: Do autor, 2014

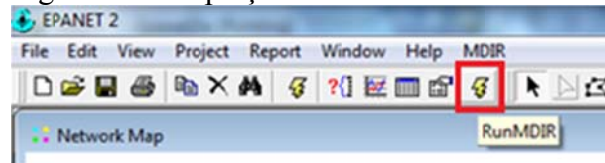
4.2 ADAPTAÇÃO NO AMBIENTE DELPHI

A programação existente no ambiente Delphi serve para construir graficamente a rede de distribuição e parametrizar os dados de cada elemento (tubos, nós, etc). Os cálculos hidráulicos para a resolução do sistema são feitos somente na DLL. O que o aplicativo Delphi faz é chamar esta função externa para executar os cálculos. Como a construção da rede é mantida de forma idêntica entre os modelos Gradiente e MDIR, as alterações feitas no ambiente Delphi foram muito simples.

A alteração básica efetuada foi a duplicação do código que executa as chamadas dos cálculos na DLL e por sua vez renomeados para chamarem as rotinas implementadas para o modelo MDIR.

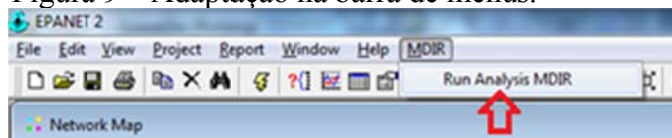
As alterações feitas no visual gráfico foram: um novo botão para executar o MDIR (Figura 8) e um novo menu para rodar a rotina do MDIR (Figura 9). Onde ambos convergem para o mesmo código fonte: FsimulMDIR.pas.

Figura 8 – Adaptação na barra de ferramentas.



Fonte: Do autor, 2014

Figura 9 – Adaptação na barra de menus.



Fonte: Do autor, 2014

Para incluir as chamadas às rotinas de cálculos do modelo MDIR, foi duplicado o código do arquivo Fsimul.pas para FsimulMDIR.pas. Onde foram realizadas as seguintes alterações básicas na rotina RunHydraulics conforme o Quadro 2.

Quadro 2: Principais alterações na rotina RunHydraulics

```

begin
  if ENopenH2() = 0 then
  begin
    ...
    ENinith2(1);
    ...
    repeat
      ...
      err := ENrunH2(t);
      tstep := 0;
      if err <= 100 then err := ENnextH2(tstep);
      ...
    until (tstep = 0) or (err > 100) or (RunStatus = rsCancelled);
  end;
  ENcloseH2();
  ...
end;

```

Fonte: Do Autor, 2014

As rotinas originais do EPANET possuem o prefixo “EN” para indicar EPANET, em seguida, figura o nome principal da função (exemplo: open) e o prefixo “H” indicando cálculo hidráulico no modelo gradiente. Para o modelo MDIR foi utilizada esta mesma notação, porém adaptando o prefixo “H” para “H2” que passa a indicar o modelo MDIR.

Nos critérios de paradas a opção RunStatus define se o usuário solicitou o cancelamento da simulação através do botão cancelar.

Para ser possível as chamadas das funções externas da DLL foi necessário alterar o arquivo epanet2.pas.

Neste arquivo, foram declaradas as funções externas (veja Quadro 3) na seção *interface* visto que esta seção é utilizada para informar ao compilador quais objetos e variáveis utilizadas na *unit* são acessíveis por outras *units* (WARNER e GOLDSMAN, 1996, p. 66).

Quadro 3: Declaração de funções externas na seção interface (epanet2.pas)

```
function ENopenH2: Integer; stdcall;
function ENinitH2(SaveFlag: Integer): Integer; stdcall;
function ENrunH2(var T: LongInt): Integer; stdcall;
function ENnextH2(var Tstep: LongInt): Integer; stdcall;
function ENcloseH2: Integer; stdcall;
```

Fonte: Do Autor, 2014

Por serem funções na linguagem C, as mesmas foram declaradas com `stdcall`, pois: “As funções exportadas também dever ser declaradas como *stdcall* para usar a técnica de passagem de parâmetros padrão do Win32, em vez da técnica de passagem de parâmetro de registro otimizada (que é o padrão no Delphi).[...]”. (CANTÙ, 2003, p. 322)

Na seção *implementation*, é onde o compilador irá buscar pelas instruções reais para executar no computador (WARNER e GOLDSMAN, 1996, p. 67). Observa-se no Quadro 4, que o código deve ser buscado em um arquivo externo (DLL).

Quadro 4: Declaração de funções externas na seção implementation (epanet2.pas)

```
function ENopenH2;           external 'EPANET2MDIR.DLL';
function ENinitH2;          external 'EPANET2MDIR.DLL';
function ENrunH2;           external 'EPANET2MDIR.DLL';
function ENnextH2;          external 'EPANET2MDIR.DLL';
function ENcloseH2;         external 'EPANET2MDIR.DLL';
```

Fonte: Do Autor, 2014

4.3 TÉCNICAS DE DEPURAÇÃO NO DEV-C++

O processo de depuração foi utilizado com o objetivo de se aprender o funcionamento do motor de cálculo do software EPANET. Optou-se pela utilização da IDE DEV-C++ pelo fato deste ambiente de desenvolvimento já ser de conhecimento do autor.

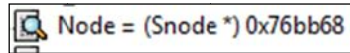
Algumas técnicas de depuração tiveram que ser pesquisadas visto que o manual do DEV-C++ não traz as opções de depuração bem documentadas.

Horstmann (2005, p. 321) destaca que “Vetores são mais difíceis de inspecionar no depurador do que números ou objetos.[...]”. O autor apresenta algumas técnicas para depurar vetores que através de testes e tentativas com vetores do EPANET, conseguiu-se chegar a alguns métodos, que serão explicados abaixo.

No depurador do DEV-C++, utilizando o comando adicionar marcador para informar o nome de variável que se deseja depurar, pôde-se observar diversos resultados.

Utilizando Node (variável que armazena os dados dos nós - pontos de consumo de água - em forma de registro), o depurador indicou que é uma variável ponteiro com o valor 0x76bb68 do tipo Snode (estrutura definida no EPANET para armazenar os dados dos nós), veja Figura 10.

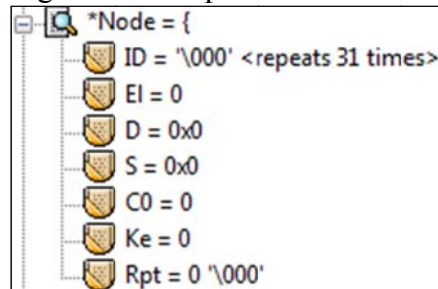
Figura 10 – Depuração: Node



Fonte: Do autor, 2014

Para depurar o conteúdo armazenado nesta estrutura, utiliza-se a notação padrão da linguagem C bastando acrescentar o sinal * antes no nome do ponteiro. Veja Figura 11.

Figura 11 – Depurando elemento inicial (1ª forma)

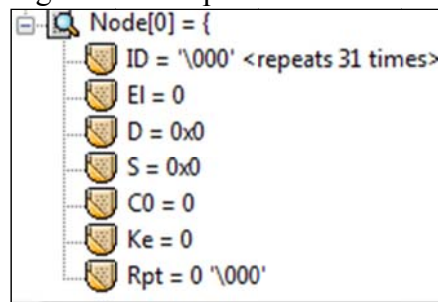


Fonte: Do autor, 2014

Observe que foi exibido os valores do 1º elemento, sendo que o exemplo testado é composto de um vetor com 11 elementos. Onde cada elemento do vetor é uma estrutura Snode. Pode-se afirmar que temos um vetor de estrutura com 11 elementos.

Para depurar o conteúdo armazenado no primeiro elemento pode-se utilizar outra forma veja Figura 12, onde é retirado o * e colocado ao final entre colchetes o elemento que se deseja visualizar. Como os vetores iniciam sua numeração a partir do zero, este será o primeiro elemento.

Figura 12 – Depurando elemento inicial (2ª forma)

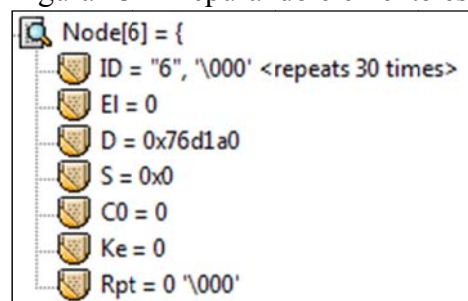


Fonte: Do autor, 2014

Neste ponto cabe fazer uma observação sobre a forma de armazenamento dos vetores no software EPANET, pois, em todos os vetores o primeiro elemento é inutilizado, isto é, são deixados com valores nulos ou zeros e não são contabilizados nos algoritmos do EPANET. Esta técnica foi utilizada para que nos loops dos algoritmos os programadores usassem uma contagem natural como de 1 a 11 se fosse uma rede com 11 nós. Assim, o vetor terá 12 elementos sendo que o elemento 0 é ignorado e os demais seguem uma contagem natural: o elemento 12º é o elemento 11 do vetor, pois a contagem começa do zero.

Veja na Figura 13 como fazer a exibição do 6º elemento do vetor.

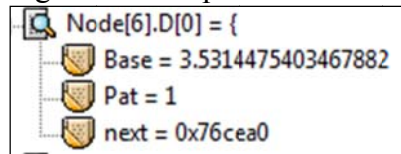
Figura 13 – Depurando elemento específico



Fonte: Do autor, 2014

Agora se pode observar que o nó 6 possui uma lista de demandas (consumo) cadastradas em seu nó. Veja que a variável D aponta para uma área de memória e a variável S não. Para visualizar o conteúdo desta área de memória apontada, deve-se escrever a variável estrutura com sua posição entre colchetes, em seguida a variável D com sua posição zero (posição inicial) entre colchetes veja Figura 14.

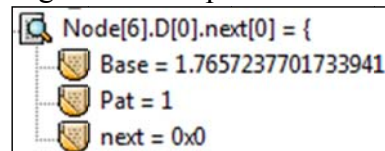
Figura 14 – Depurando uma lista na estrutura (1º elemento)



Fonte: Do autor, 2014

Verificando que este registro possui uma variável chamada *next* que também é um ponteiro pode-se deduzir que está sendo utilizada uma lista encadeada e para localizar o 2º elemento será necessário apontar para o endereço desta variável *next*, veja Figura 15.

Figura 15 – Depurando uma lista na estrutura (2º elemento)



Fonte: Do autor, 2014

Uma tentativa que poder-se-ia imaginar, seria tentar visualizar este mesmo registro usando a seguinte notação: Node[6].D[1]. Porém esta tentativa será frustrada, pois se trata de uma lista encadeada que não está numa ordem sequencial na memória do computador. Este tipo de notação poderia ser usado se o elemento D fosse um vetor de um tipo simples (como inteiro ou float) e alocado na quantidade de elementos necessários. Existem estruturas no EPANET que utilizam este método e neles sim, pode-se usar esta notação direta.

Na Figura 16, se faz uma tentativa de visualizar o 3º elemento, e como é um elemento que não existe, pois o último *next* está como nulo (não aponta para nenhuma área de memória), o depurador indica a impossibilidade de sua visualização.

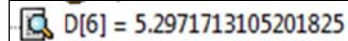
Figura 16 – Depurando uma lista na estrutura (3º elemento - inexistente)



Fonte: Do autor, 2014

Para os próximos exemplos será utilizado um vetor D com 11 posições onde o EPANET armazena a demanda de cada nó em suas respectivas posições. Para visualizar a demanda no nó 6, basta usar D[6], conforme Figura 17.

Figura 17 – Depurando elemento do vetor específico

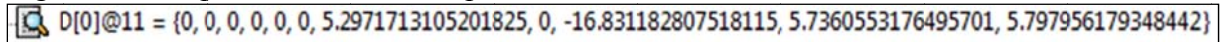


```
D[6] = 5.2971713105201825
```

Fonte: Do autor, 2014

Nestes casos a opção de visualizar todos os valores de cada posição é o ideal para acompanhar as alterações de dados que vão sendo efetuadas à medida que se caminha pelo programa. Para obter esta visualização, basta visualizar o primeiro elemento com `D[0]` e indicar que se deseja deslocar a visualização para os próximos 11 elementos acrescentando o sinal `@` e o número de elementos desejados, veja Figura 18.

Figura 18 – Depurando o vetor completo



```
D[0]@11 = {0, 0, 0, 0, 0, 0, 5.2971713105201825, 0, -16.831182807518115, 5.7360553176495701, 5.797956179348442}
```

Fonte: Do autor, 2014

4.4 ADAPTAÇÃO NA DLL

A biblioteca externa `epanet2.dll` onde os cálculos hidráulicos são efetuados foi escrita na linguagem C. Os arquivos alterados foram: `epanet.c`, `hydraul2.c` e `vars.h`.

O arquivo `hydraul2.c` foi criado da duplicação do arquivo `hydraul.c` que realiza os cálculos do método gradiente e cujo arquivo não foi alterado, manteve-se o cálculo original do EPANET. Por sua vez o arquivo `hydraul2.c` passou a ser adaptado para receber os cálculos do modelo MDIR. Desta forma o software final realiza os 2 cálculos: método gradiente e método MDIR bastando o usuário escolher o item desejado.

O arquivo `epanet.c` é o módulo principal. É dele que se chama as demais funções do arquivo `hydraul.c` (método gradiente) e `hydraul2.c` (método MDIR). Por este motivo neste módulo principal foi necessário duplicar as funções de chamadas das rotinas hidráulicas e adaptá-las para chamar as rotinas do MDIR conforme Quadro 5, onde as alterações entre a rotina original e a duplicada se encontram em negrito para melhor visualização das diferenças. Todas as funções duplicadas neste arquivo `epanet.c` tiveram somente estas alterações simplificadas. Desta forma as grandes alterações na forma de cálculo foram realizadas e centralizadas no arquivo `hydraul2.c`. Nota-se também que foi utilizado o sufixo 2 para indicar rotinas de cálculos para o MDIR.

Quadro 5: Exemplo de adaptação no arquivo epanet.c

```

int DLLEXPORT ENcloseH()
{
    if (!Openflag) return(102);
    closehyd();
    OpenHflag = FALSE;
    return(0);
}

int DLLEXPORT ENcloseH2()
{
    if (!Openflag) return(102);
    closehyd2();
    OpenHflag = FALSE;
    return(0);
}

```

Fonte: Do Autor, 2014

No arquivo vars.h, foram criados ponteiros de matrizes que são usadas no arquivo hydraul2.c para o modelo MDIR, conforme Quadro 6.

Quadro 6: Ponteiros de matrizes criadas em vars.h

```

//matriz de 1 dimensao
EXTERN PGMATRIX ParcA,
                ParcC,
                ...
                ParcZ;
//matriz de 2 dimensoes
EXTERN PGMATRIX C21BInv,
                ...
                BInvC10;

```

Fonte: Do Autor, 2014

Conforme descrito anteriormente, o recurso de depuração foi muito utilizado para a realização da engenharia reversa.

4.5 GMATRIX

Para a inserção dos códigos referentes aos cálculos do modelo MDIR era necessário a realização de diversos cálculos matriciais. Inicialmente começou-se a desenvolver algoritmos (funções) para efetuarem estes cálculos. No entanto, pesquisando na internet foi encontrada uma biblioteca que já disponibilizava estas funções cujo nome é GMATRIX.

A biblioteca GMATRIX foi desenvolvida por Geovany Araújo Borges, professor adjunto pelo Departamento de Engenharia Elétrica da Universidade de Brasília (UnB), onde coordena o Laboratório de Automação e Robótica (LARA).

A biblioteca GMATRIX pode ser vista como uma alternativa para a implementação de matrizes em linguagem C, em que o tipo matriz é uma estrutura de dados. Ela vem sendo aperfeiçoada há pelo menos cinco anos, tendo sido usada com sucesso em vários projetos de robótica, em que a precisão e a segurança se fazem necessários. Várias rotinas de cálculo matricial e álgebra linear estão incluídas na biblioteca, além de interface MATLAB e suporte para microcontroladores. (BORGES, 2005)

Após estudo do manual do GMATRIX vale destacar como ficou a forma de trabalho utilizando esta biblioteca. Primeiramente, observa-se no Quadro 7, que matrizes de 1 ou 2 dimensões são declaradas da mesma forma, pois o que irá diferenciá-las será no momento da alocação de memória.

Quadro 7: Declaração de ponteiros do tipo PGMATRIX

```
//matriz de 1 dimensao
EXTERN PGMATRIX ParcZ;

//matriz de 2 dimensoes
EXTERN PGMATRIX BInvC10;
```

Fonte: Do Autor, 2014

No Quadro 8, observa-se que a alocação da matriz é feita utilizando-se uma função já definida chamada PGMATRIX_ALLOC. Onde se define o tamanho das linhas e das colunas em seus argumentos. Pode-se verificar que uma matriz de 1 dimensão é alocada como uma matriz de 2 dimensões, sendo n linhas X 1 coluna.

Quadro 8: Alocação dos ponteiros do tipo PGMATRIX

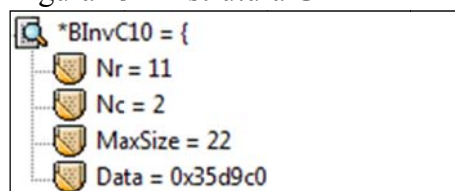
```
int allocmatrix2()
{
    ...
    //matriz de 1 dimensao
    ParcZ = PGMATRIX_ALLOC(Nlinks, 1);
    ...
    //matriz de 2 dimensoes
    BInvC10 = PGMATRIX_ALLOC(Nlinks, Ntanks);
    ...
}
```

Fonte: Do Autor, 2014

Cabe esclarecer que a letra P da função PGMATRIX_ALLOC, indica que se deseja trabalhar com ponteiros, pois existe outra função GMATRIX_DECLARE que trabalha com matrizes como variáveis e não ponteiros. Como há a necessidade de várias funções acessarem os dados das matrizes que vão sendo gerados, este trabalho se torna facilitado com ponteiros.

A estrutura do ponteiro GMATRIX é composta pelas variáveis: Nr (número de linhas), Nc (número de colunas) e MaxSize (tamanho máximo) utilizado para evitar estouro da matriz. A variável Data é um ponteiro que indica o início dos dados, veja Figura 19.

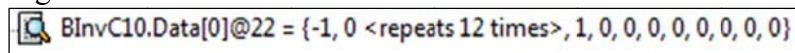
Figura 19 – Estrutura GMATRIX



Fonte: Do autor, 2014

Para visualizar os dados da matriz basta utilizar a técnica descrita anteriormente deslocando 22 elementos a partir do primeiro elemento indicado por [0], veja Figura 20.

Figura 20 – Visualizando os dados da matriz



Fonte: Do autor, 2014

Note que os dados não são visualizados com uma tabela de 11 linhas por 2 colunas: estes posicionamentos são feitos via algoritmos do GMATRIX. O armazenamento final é feito num vetor de 22 elementos.

Para a utilização da biblioteca GMATRIX, foi solicitada, via e-mail, autorização para seu uso no software EPANET. A autorização está no Anexo A – Autorização para uso do GMATRIX

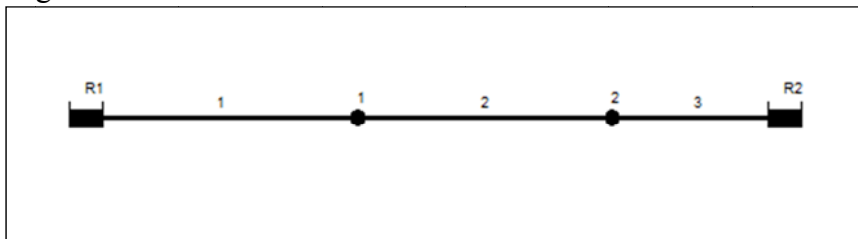
5 RESULTADOS COMPARATIVO DO MDIR COM O MÉTODO GRADIENTE

Após fazer a implementação do método MDIR no software EPANET, buscou-se a realização de teste, confrontando o método gradiente com método MDIR para verificar se os resultados estavam próximos e válidos.

5.1 SIMULAÇÃO REDE 01 SEM DEMANDA

A primeira rede testada utilizou-se de reservatórios, nós e tubos conforme Figura 21.

Figura 21 – Leiaute da rede 01



Fonte: Do autor, 2014

O sistema hidráulico está composto de um reservatório R1 com nível constante igual a 10m, 3 tubulações de 300m de comprimento cada, diâmetro de 400mm e um reservatório R2 com nível constante igual a 0m. Os nós não possuem elevações e nem demandas.

Espera-se como resultado desta simulação que o método gradiente se acomode repentinamente à vazão de água determinada pela pressão do reservatório e que o método MDIR faça uma transição gradual do estado de repouso para a vazão final.

As principais configurações de tempos a serem observadas neste teste são: Hydraulic Time Step (etapa de tempo hidráulico) onde a duração de cada etapa irá ser de 2 segundos e Reporting Time Step (etapa de tempo do relatório) que também recebeu a mesma configuração, conforme Figura 22.

Figura 22 – Configuração dos tempos de cálculos da rede 01

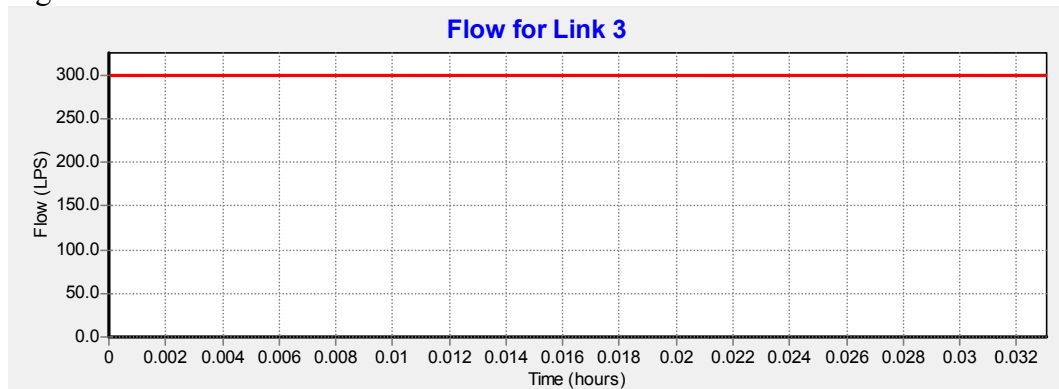
Property	Hrs:Min
Total Duration	0:02
Hydraulic Time Step	0:00:2
Quality Time Step	0:05
Pattern Time Step	6
Pattern Start Time	0:00
Reporting Time Step	0:00:2
Report Start Time	0:00
Clock Start Time	12 am
Statistic	None

Fonte: Do autor, 2014

Inicialmente tentou-se realizar os testes com a duração de 1 segundo por etapa, porém o EPANET não executava corretamente. Fazendo a depuração do programa percebeu-se que o EPANET decrece 1 unidade na variável que controla o tempo, assim utilizando 2 segundos internamente o EPANET estará usando 1 segundo.

Como resultado desta simulação através do método gradiente, observa-se que a vazão já se adequa abruptamente na vazão inicial, pois este modelo não aplica cálculos da inércia, veja Figura 23.

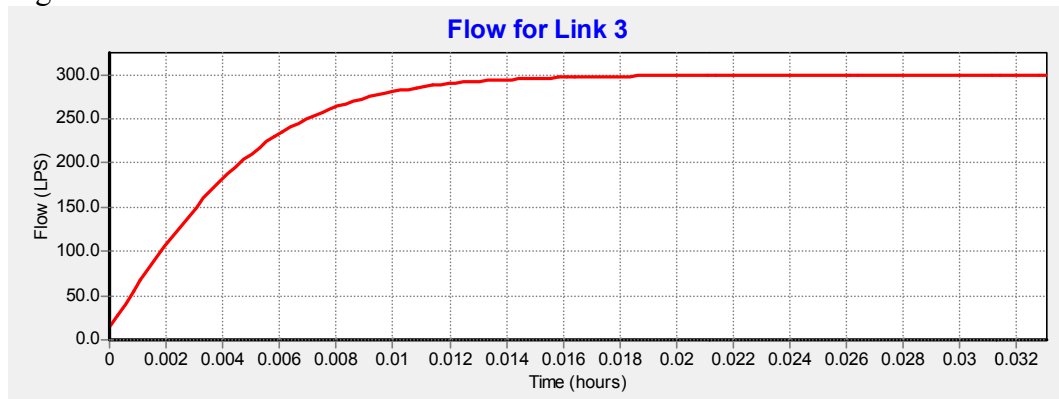
Figura 23 – Vazão do tubo 3 - Gradiente



Fonte: Do autor, 2014

Por outro lado, no método MDIR, observa-se que a vazão já se adequa lentamente à vazão inicial, pois este modelo aplica cálculos da inércia, veja Figura 24.

Figura 24 – Vazão do tubo 3 - MDIR



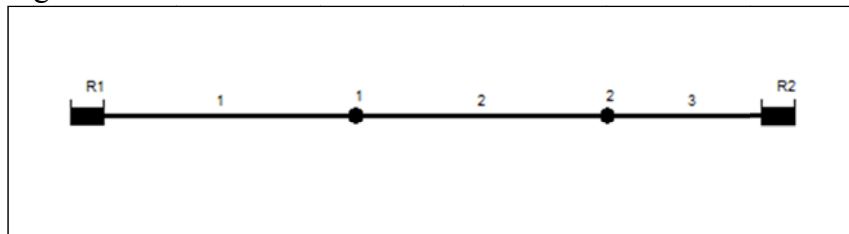
Fonte: Do autor, 2014

O resultado esperado foi alcançado visto que no método gradiente a vazão de água determinada pela pressão do reservatório foi alcançada repentinamente e no método MDIR houve uma transição gradual do estado de repouso para a vazão final.

5.2 SIMULAÇÃO REDE 02

A segunda rede possui o mesmo leiaute físico que a rede 01, conforme Figura 25.

Figura 25 – Leiaute da rede 02



Fonte: Do autor, 2014

O sistema hidráulico está composto de um reservatório R1 com nível constante igual a 80m, 3 tubulações de 300m de comprimento cada, diâmetro de 400mm e um reservatório R2 com nível constante igual a 70m. Os nós não possuem elevações e nem demandas. Porém o nó 1 que iniciará sem demanda, irá receber posteriormente uma demanda de 100L/S.

Espera-se como resultado desta simulação que o método gradiente registre a diminuição de pressão e o aumento da vazão devido ao aumento do consumo (demanda). No modelo MDIR se espera as mesmas situações, porém com a indicação de picos de pressão e a adaptação gradual da vazão ao realizar a mudança de estados de consumo.

Na Figura 26, observa-se a configurações de tempos utilizadas.

Figura 26 – Configuração dos tempos de cálculos rede 02

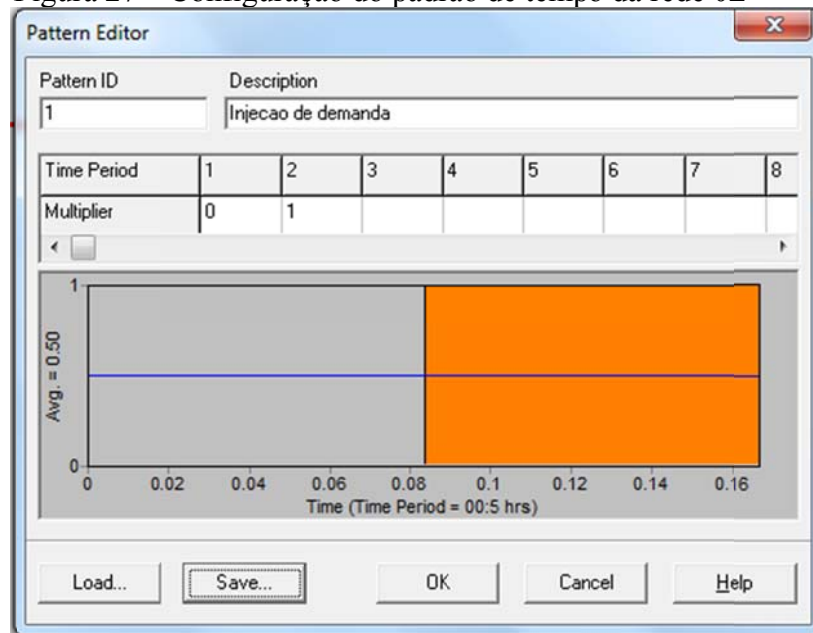


Property	Hrs:Min
Total Duration	00:20
Hydraulic Time Step	0:00:2
Quality Time Step	0:05
Pattern Time Step	00:5
Pattern Start Time	0:00
Reporting Time Step	0:00:2
Report Start Time	0:00
Clock Start Time	12 am
Statistic	None

Fonte: Do autor, 2014

Para injetar a demanda no nó 1 foi utilizado o recurso de Pattern (padrões) que neste caso específico será um padrão de tempo. Observa-se na Figura 26 que o Pattern Time Step (etapa de tempo do padrão) está configurado para 5 minutos. Esta mesma configuração pode ser observada na configuração do padrão na Figura 27 onde aparece o Time Period (período de tempo) de 00:5 hrs que é o equivalente a 5 minutos. Na grade de multiplicação foram usados somente 2 períodos (cada um será de 5 minutos), onde foram inseridos os valores: 0 e 1. Isto funcionará da seguinte forma: no 1º período de 5 minutos a demanda de 100L/S será multiplicada por 0 tornando-se assim uma demanda igual a 0, no 2º período será 100×1 que resultará na própria demanda de 100L/S. Este ciclo volta a se repetir até terminar todo o processo.

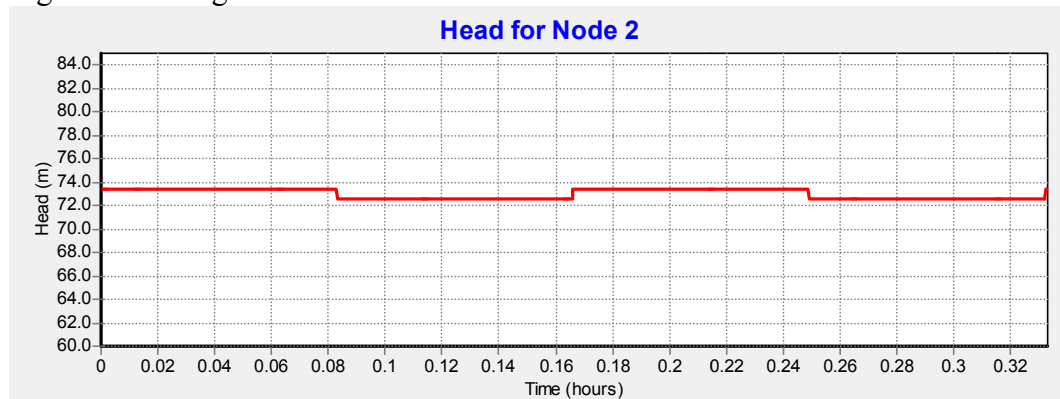
Figura 27 – Configuração do padrão de tempo da rede 02



Fonte: Do autor, 2014

Como resultado desta simulação através do método gradiente, observa-se que a carga hidráulica se adequa de um nível para o outro sem registrar picos de pressão, pois este modelo não aplica cálculos da inércia, veja Figura 28.

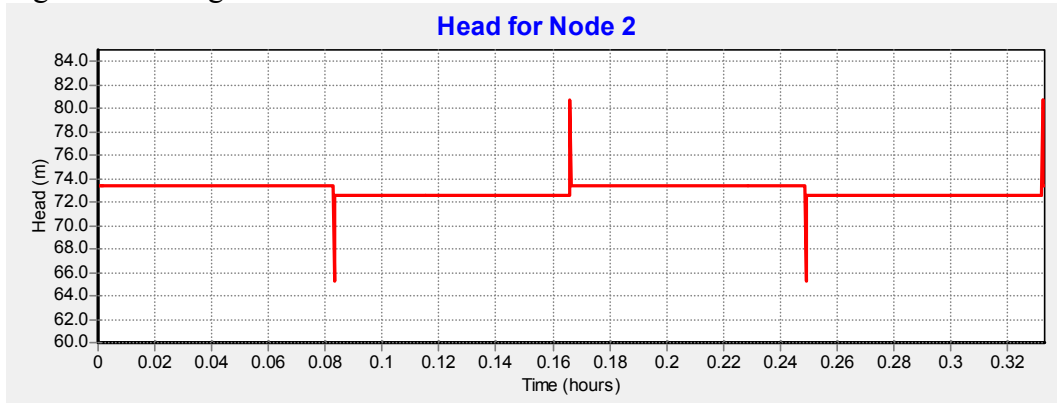
Figura 28 – Carga hidráulica do nó 2 - Gradiente



Fonte: Do autor, 2014

Por outro lado, no método MDIR, observa-se que a carga hidráulica registra os picos de pressão, pois este modelo aplica cálculos da inércia, veja Figura 29.

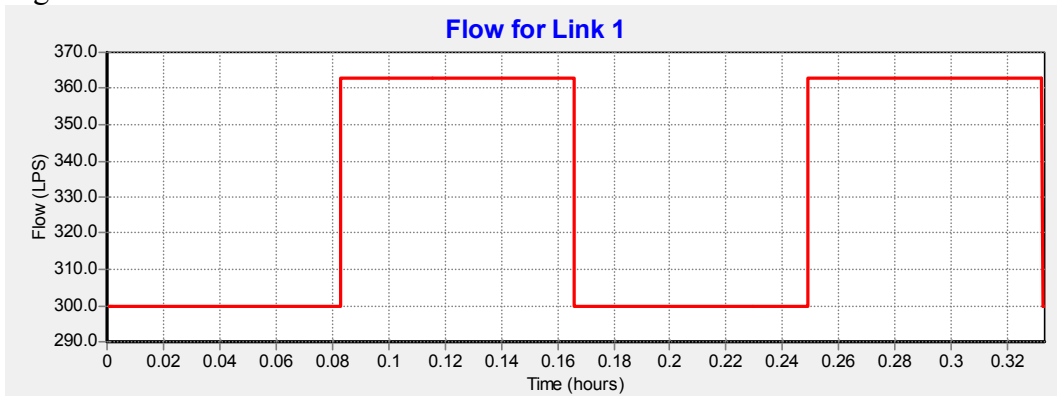
Figura 29 – Carga hidráulica do nó 2 - MDIR



Fonte: Do autor, 2014

Veja na Figura 30 a vazão no tubo a cada inserção e retirada de demanda no método gradiente.

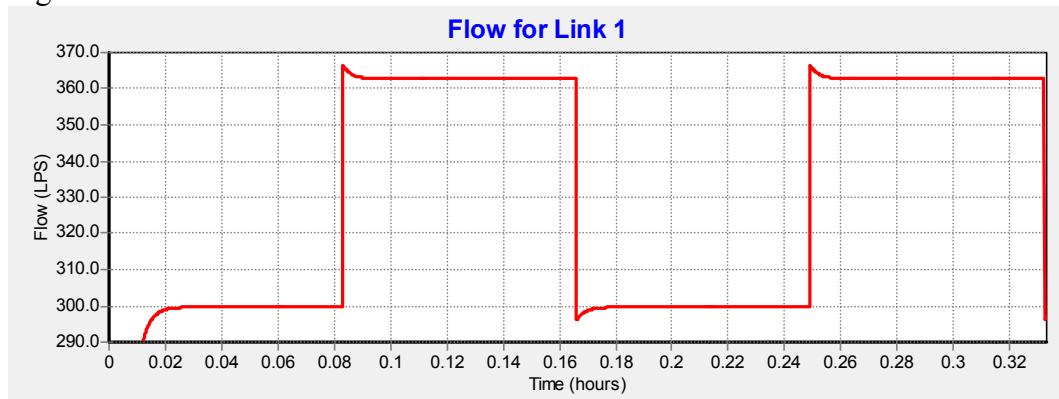
Figura 30 – Vazão do tubo 1 - Gradiente



Fonte: Do autor, 2014

E na Figura 31 o método MDIR.

Figura 31 – Vazão do tubo 1 - MDIR



Fonte: Do autor, 2014

Neste gráfico observa-se que o modelo MDIR gasta um tempo inicial para se adaptar à vazão pois o mesmo é inicializado com 0,000001. Este período inicial deve então ser desprezado. Para retirar esta parte do gráfico pode-se configurar o Report Start Time (Tempo Inicial do Relatório) para o tempo que se deseja cortar, veja Figura 32.

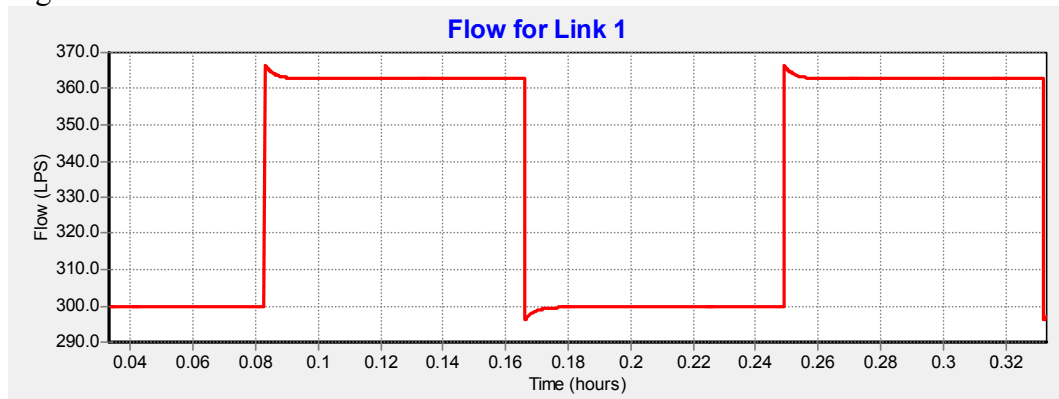
Figura 32 – Retirando tempo inicial do gráfico da rede 02

Times Options	
Property	Hrs:Min
Total Duration	00:20
Hydraulic Time Step	0:00:2
Quality Time Step	0:05
Pattern Time Step	00:5
Pattern Start Time	0:00
Reporting Time Step	0:00:2
Report Start Time	0:02
Clock Start Time	12 am
Statistic	None

Fonte: Do autor, 2014

Na Figura 33, verifica-se o resultado do corte.

Figura 33 – Vazão do tubo 1 – MDIR – cortando instantes iniciais



Fonte: Do autor, 2014

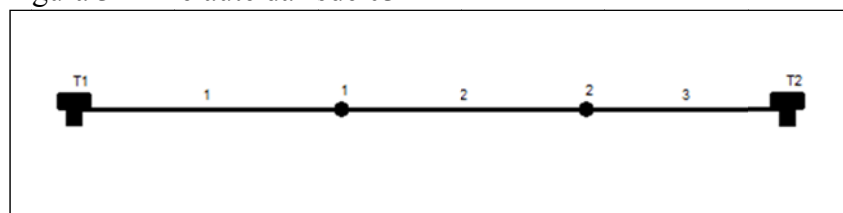
Daqui em diante todo gráfico que necessitar retirar a parte inicial já será feita utilizando a explicação citada acima.

O resultado esperado foi alcançado visto que no método gradiente foi registrada a diminuição de pressão e o aumento da vazão devido ao aumento do consumo (demanda) e vice-versa. E no método MDIR houve as mesmas situações, porém com a indicação de picos de pressão e a adaptação gradual da vazão ao realizar a mudança de estados de consumo.

5.3 SIMULAÇÃO REDE 03

A terceira rede possui o leiaute conforme a Figura 34.

Figura 34 – Leiaute da rede 03



Fonte: Do autor, 2014

O sistema hidráulico está composto de um tanque T1 com nível variável e nível inicial igual a 80m, um tanque T2 com nível variável e nível inicial igual a 70m, ambos com diâmetro igual a 3m. Possuem ainda 3 tubulações de 300m de comprimento cada, diâmetro de 400mm. Os nós não possuem elevações e nem demandas.

Espera-se como resultado desta simulação que o método gradiente registre a diminuição de pressão do T1 de forma gradual, mas sem oscilação e a estabilização da vazão

no estado de repouso de forma gradual também sem oscilação. No modelo MDIR se espera as mesmas situações, porém com oscilações na diminuição da pressão do T1 e na estabilização da vazão no estado de repouso devido ao fator da inércia.

Na Figura 35, observa-se a configurações de tempos utilizadas.

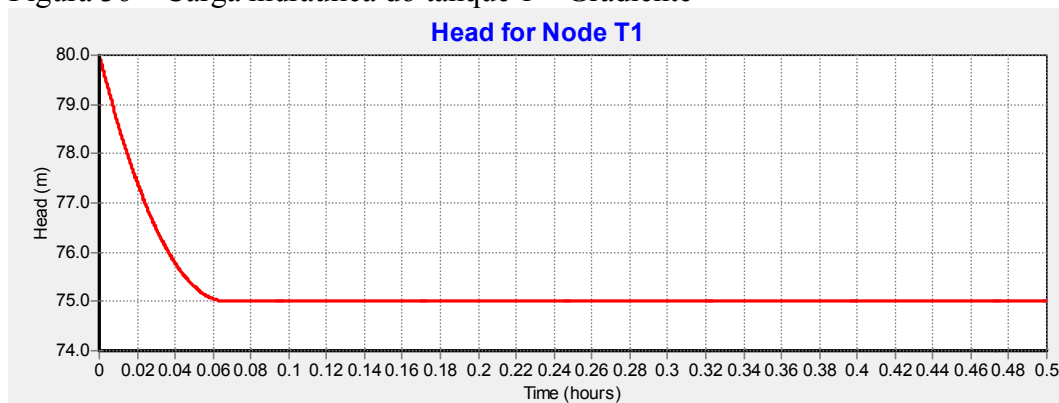
Figura 35 – Configuração dos tempos de cálculos da rede 03

Property	Hrs:Min
Total Duration	00:30
Hydraulic Time Step	0:00:2
Quality Time Step	0:05
Pattern Time Step	6
Pattern Start Time	0:00
Reporting Time Step	0:00:2
Report Start Time	0:00
Clock Start Time	12 am
Statistic	None

Fonte: Do autor, 2014

Na Figura 36 observa-se a carga hidráulica resultante do método gradiente.

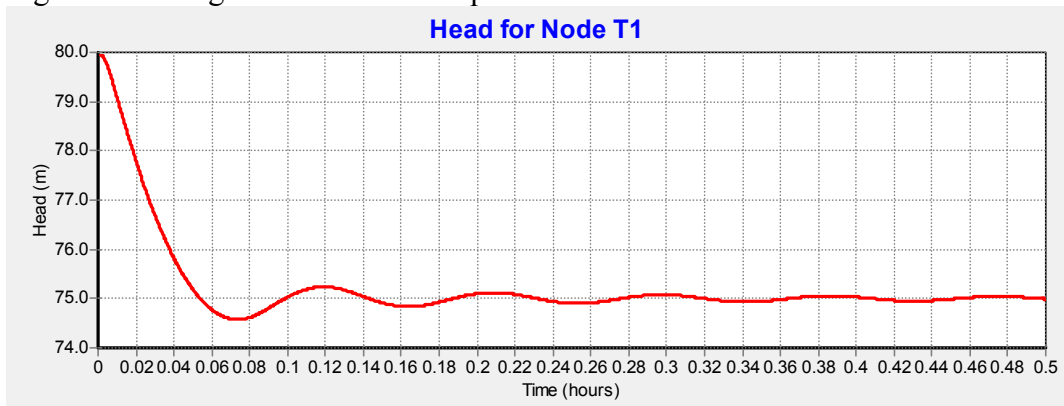
Figura 36 – Carga hidráulica do tanque 1 – Gradiente



Fonte: Do autor, 2014

E na Figura 37 observa-se a carga hidráulica resultante do método MDIR. Onde a oscilação caracteriza o efeito da inércia conhecido com pêndulo hidráulico “onde um certo desnível de água entre dois reservatórios é abandonado para atingir uma condição de regime permanente final quando eles atingem o mesmo nível.” (ANJO, 2008, p. 137) .

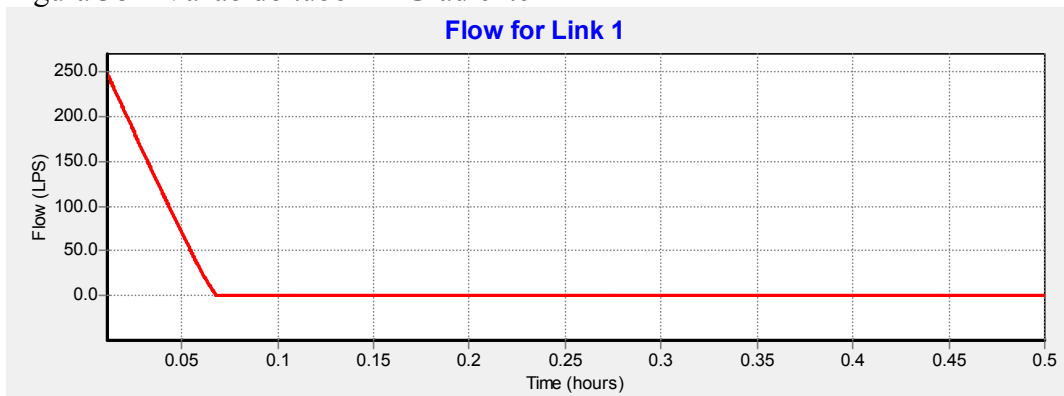
Figura 37 – Carga hidráulica do tanque 1 – MDIR



Fonte: Do autor, 2014

Na Figura 38 observa-se a vazão resultante do método gradiente.

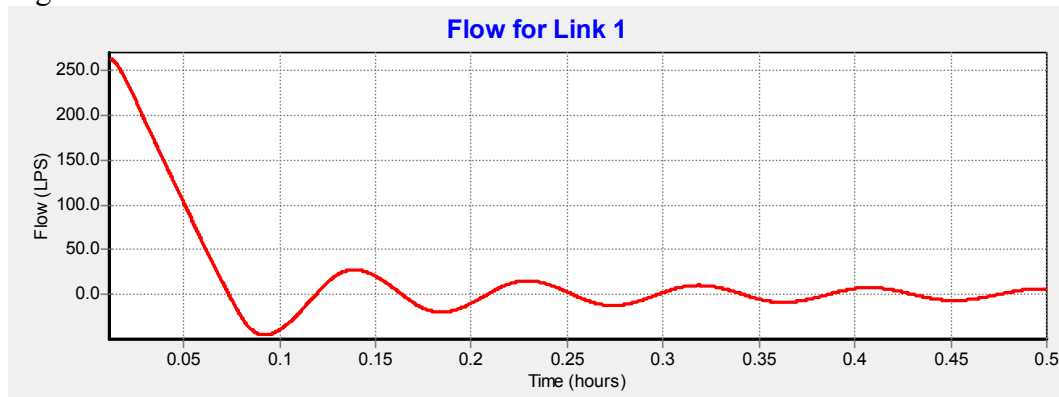
Figura 38 – Vazão do tubo 1 – Gradiente



Fonte: Do autor, 2014

Na Figura 39 observa-se a vazão resultante do método MDIR e novamente a oscilação característica do pêndulo hidráulico.

Figura 39 – Vazão do tubo 1 – MDIR



Fonte: Do autor, 2014

O resultado esperado foi alcançado visto que no método gradiente foi registrada a diminuição de pressão do T1 de forma gradual e a estabilização da vazão no estado de repouso de forma gradual, ambas sem oscilações. E no método MDIR houve as mesmas situações, porém com a oscilação de pressão em torno de 75 m e a oscilação de vazão em torno do estado de repouso 0 LPS, caracterizando assim a inversão de sentido devido ao efeito do pêndulo hidráulico.

5.4 SIMULAÇÃO REDE 04

A quarta rede possui o leiaute físico idêntico á rede 03, porém as 3 tubulações passaram a ter 30m de comprimento cada e diâmetro de 600mm. Os nós continuaram sem elevações e sem demandas.

Espera-se como resultado desta simulação os mesmos resultados previstos na simulação 3 porém com maior intensidade devido ao ajustes do comprimento e diâmetro das tubulações que proporcionam um efeito maior da inércia.

Na Figura 40, observa-se a configurações de tempos utilizadas.

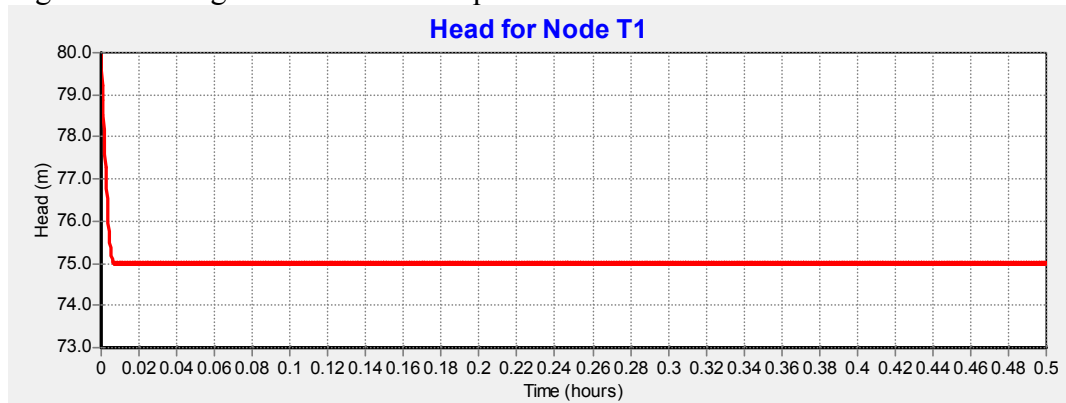
Figura 40 – Configuração dos tempos de cálculos da rede 04

Property	Hrs:Min
Total Duration	00:30
Hydraulic Time Step	0:00:2
Quality Time Step	0:05
Pattern Time Step	6
Pattern Start Time	0:00
Reporting Time Step	0:00:2
Report Start Time	0:00
Clock Start Time	12 am
Statistic	None

Fonte: Do autor, 2014

Na Figura 41 observa-se a carga hidráulica resultante do método gradiente.

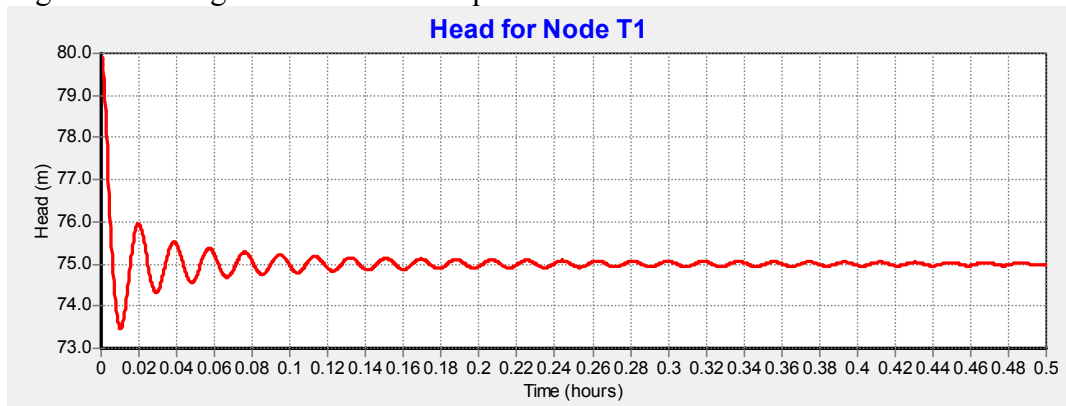
Figura 41 – Carga hidráulica do tanque 1 – Gradiente



Fonte: Do autor, 2014

E na Figura 42 observa-se a carga hidráulica resultante do método MDIR. Onde a oscilação característica do efeito da inércia passa a ser bem mais acentuada comparada à Figura 37.

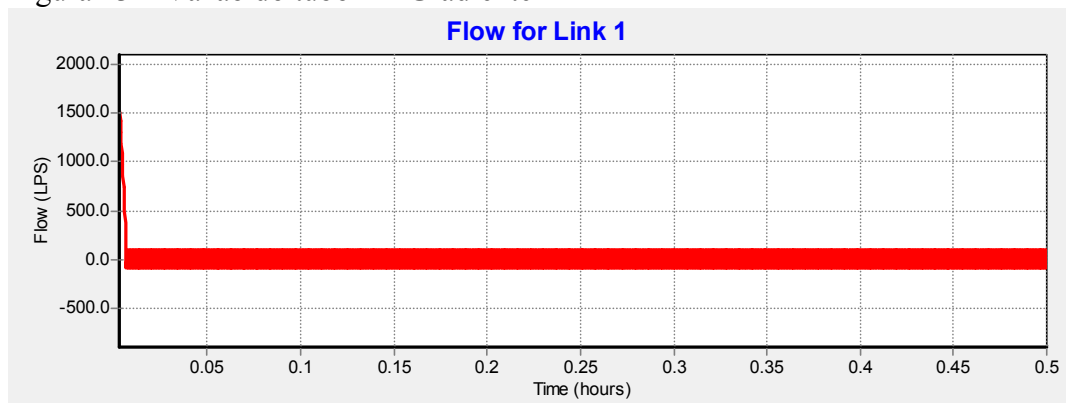
Figura 42 – Carga hidráulica do tanque 1 – MDIR



Fonte: Do autor, 2014

Na Figura 43 observa-se a vazão resultante do método gradiente.

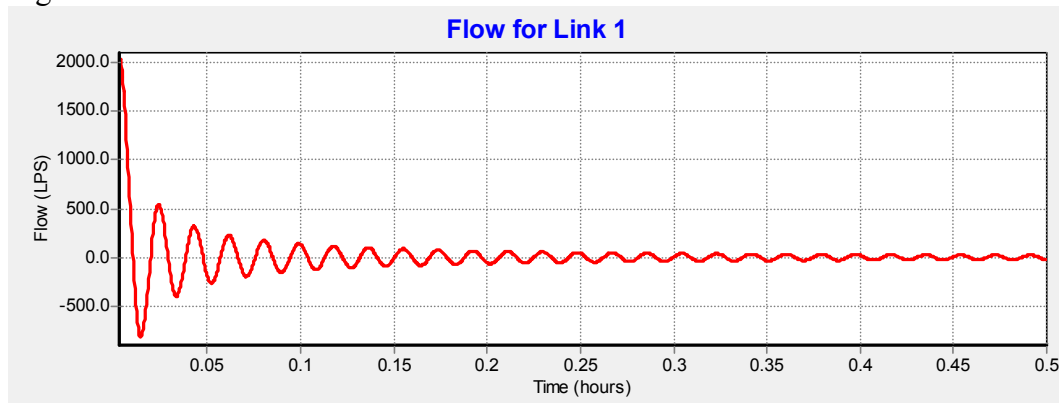
Figura 43 – Vazão do tubo 1 – Gradiente



Fonte: Do autor, 2014

Na Figura 44 observa-se a vazão resultante do método MDIR e novamente a oscilação mais acentuada da característica do pêndulo hidráulico se comparada à Figura 39.

Figura 44 – Vazão do tubo 1 – MDIR



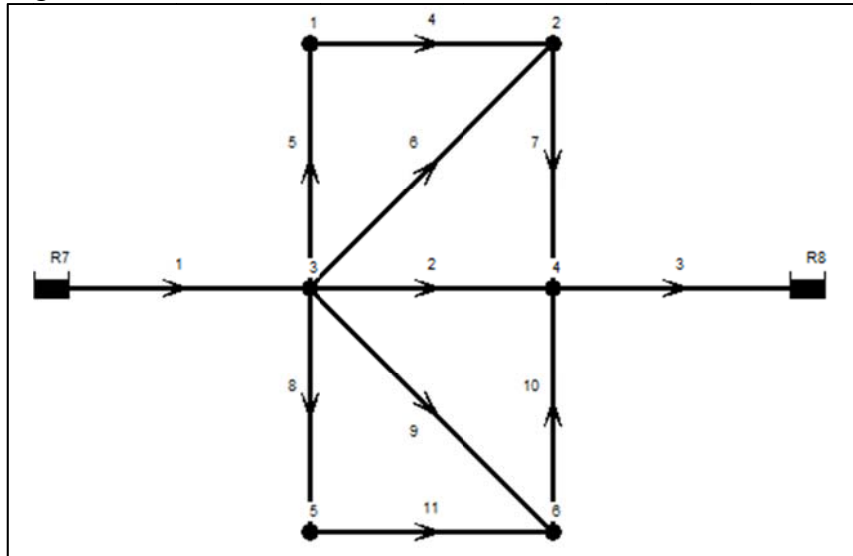
Fonte: Do autor, 2014

O resultado esperado foi alcançado visto que se obteve os mesmos resultados previstos na simulação 3, porém com maior intensidade devido ao efeito da inércia nas novas configurações do tamanho e diâmetro dos tubos.

5.5 SIMULAÇÃO REDE 05

A quinta rede possui o leiaute físico, conforme mostra a Figura 45. Esta rede hidráulica é empregada por Luvizotto Jr. (1995). Esta rede é simétrica com todos os tubos de 500m de comprimento e 400mm de diâmetro. A carga nos reservatórios R7 e R8 são constantes e valem, respectivamente, 80m e 70m. Não há demandas nos nós.

Figura 45 – Leiaute da rede 05



Fonte: Do autor, 2014

Porém para representar os efeitos da inércia, os reservatórios serão substituídos por tanques com as mesmas capacidades indicadas e com 3m de diâmetros.

Espera-se como resultado desta simulação que o método gradiente não registre as oscilações características da inércia e que o modelo MDIR faça estes apontamentos.

Na Figura 46, observa-se a configurações de tempos utilizadas.

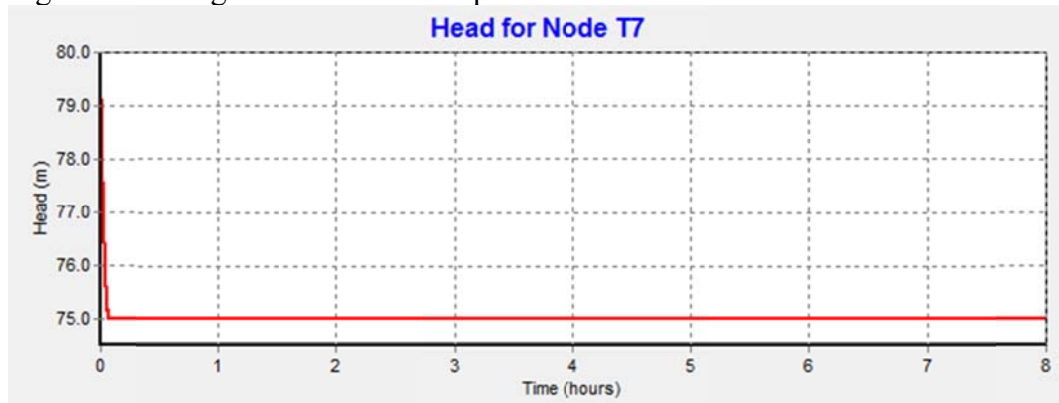
Figura 46 – Configuração dos tempos de cálculos da rede 05

Property	Hrs:Min
Total Duration	8
Hydraulic Time Step	0:00:02
Quality Time Step	0:05
Pattern Time Step	6
Pattern Start Time	0:00
Reporting Time Step	0:00:02
Report Start Time	0:00
Clock Start Time	12 am
Statistic	None

Fonte: Do autor, 2014

Na Figura 47 observa-se a carga hidráulica resultante do método Gradiente.

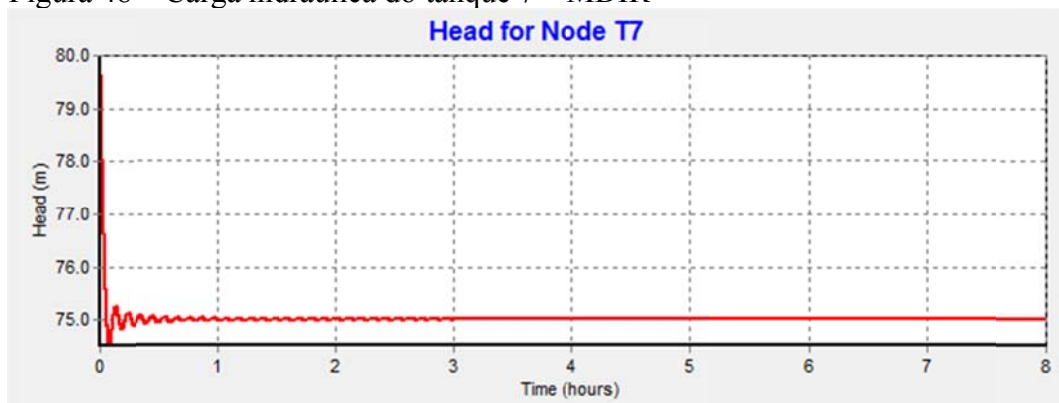
Figura 47 – Carga hidráulica do tanque 7 – Gradiente



Fonte: Do autor, 2014

Na Figura 48 observa-se a carga hidráulica resultante do método MDIR.

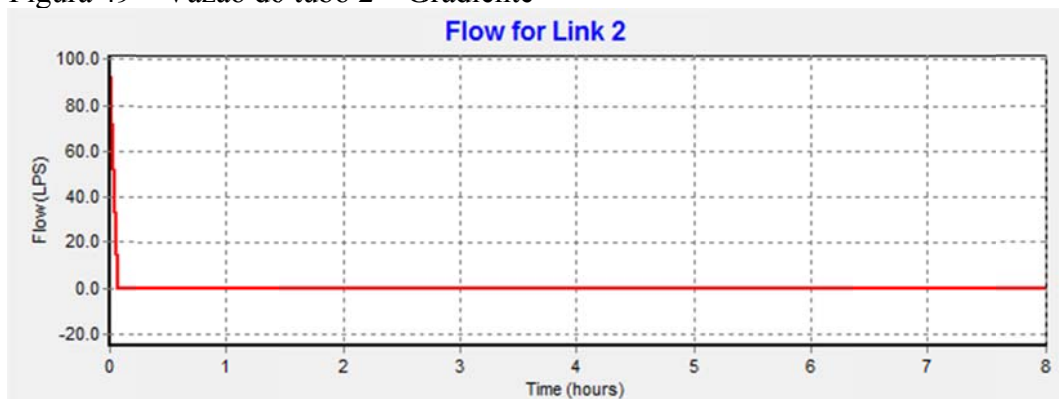
Figura 48 – Carga hidráulica do tanque 7 – MDIR



Fonte: Do autor, 2014

Na Figura 49 observa-se a vazão resultante do método Gradiente.

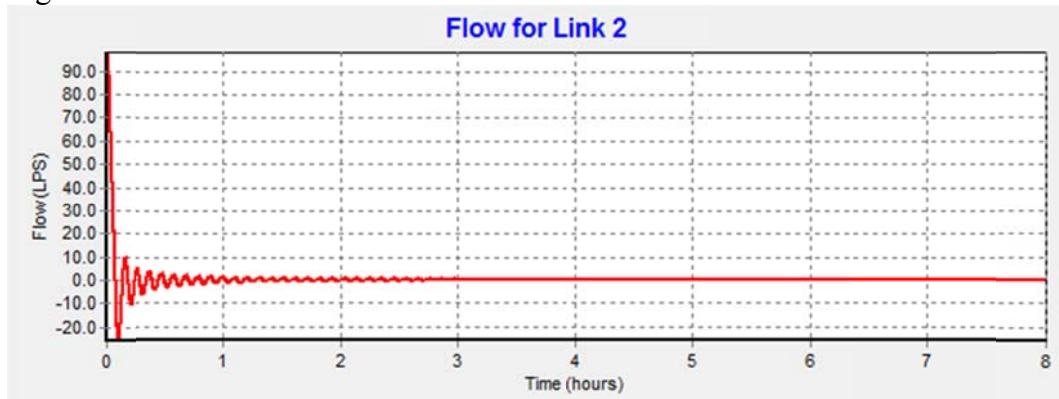
Figura 49 – Vazão do tubo 2 – Gradiente



Fonte: Do autor, 2014

Na Figura 50 observa-se a vazão resultante do método MDIR.

Figura 50 – Vazão do tubo 2 – MDIR



Fonte: Do autor, 2014

Como resultado desta simulação, o método gradiente não registrou as oscilações características da inércia e o modelo MDIR assinalou estas oscilações mesmo se tratando de uma rede hidráulica mais complexa.

6 CONCLUSÃO

Este projeto propôs a implementação de uma simulação hidráulica utilizando o Modelo Dinâmico Inercial Rígido, onde abrangeu as duas últimas etapas do abastecimento de água que são: o reservatório e a rede de distribuição. Considerou-se a água já tratada e realizou a simulação com a pressão do reservatório na rede de distribuição.

Esta simulação é importante, pois o acompanhamento da carga e da vazão nas instalações hidráulicas é necessário para detectar os níveis de abastecimento e os picos de pressão que podem ocasionar rompimentos na tubulação.

Como problema de pesquisa, foi buscado desenvolver a possibilidade de simular eventos hidráulicos na rede de distribuição levando em consideração o efeito da inércia.

O modelo hidráulico implementado (Modelo Dinâmico Inercial Rígido) leva em consideração os efeitos da inércia.

O software EPANET que é um software que simula sistemas de tubulação de distribuição de água e é de domínio público, utiliza em seus cálculos o método do gradiente onde não considera os efeitos da inércia.

Como objetivo deste trabalho, teve-se a implementação, no software EPANET, da simulação do Modelo Dinâmico Inercial Rígido (MDIR) para análise de escoamentos transitórios lentos em regime não permanente em instalações a condutos forçados e posteriormente, a comparação do modelo MDIR com o modelo Método do Gradiente (método de cálculo usado pelo EPANET).

Como metodologia, foram adquiridos os arquivos fontes do software EPANET, realizado a análise da engenharia de software, elaborado os algoritmos para o modelo MDIR e feito a pesquisa experimental comparando o Modelo MDIR com o Método do Gradiente.

O software EPANET tem como vantagens: ser de domínio público, ser utilizado em diversos trabalhos científicos, ser um dos modelos mais utilizados em diversos países, ser o mais confiável e ter compatibilidade com vários outros simuladores. No entanto apesar de ser o motor de cálculo para outros softwares comerciais, ainda possui a desvantagem de não utilizar em seus cálculos os efeitos da inércia.

Com a introdução do MDIR, no software EPANET, esta falha ficou resolvida, pois o MDIR melhorou o motor de cálculo do EPANET utilizando algoritmos que consideram os efeitos da inércia. Desta forma agora se pode detectar picos de pressão e prever rompimentos das tubulações ou de seus acessórios.

Foi utilizada a engenharia reversa como forma de estudo dos algoritmos do EPANET para identificar os componentes do sistema e suas inter-relações, aplicando-se a técnica de depuração de programas onde se é possível passar etapa por etapa no código fonte do programa. Esta depuração foi feita tanto na linguagem Delphi como na linguagem C que é onde se encontra a biblioteca externa (DLL) do motor de cálculo do EPANET.

Foram estudadas técnicas de depuração para ser possível visualizar como os dados eram armazenados pelo EPANET. E utilizado parceria com a UnB através do uso da biblioteca GMATRIX.

A ampliação do Software EPANET foi possível onde o MDIR simula o modelo dinâmico levando em consideração a inércia.

A comparação entre o método Gradiente e o método MDIR foi possível de ser realizada uma vez que o produto final permitiu a simulação em ambos os modelos, bastando o usuário escolher o método que se deseja utilizar.

A simulação no cotidiano foi aprimorada visto que além de contribuir para soluções rápidas e com custos menores, também pode prever picos de pressão e possíveis rompimentos das tubulações.

Desta forma, conclui-se que para o estudo de sistemas hidráulicos o modelo dinâmico inercial rígido (MDIR), se apresenta como uma solução muito interessante.

Sugere-se em uma próxima etapa adaptar o MDIR ao trabalho com bombas e válvulas de controle e estudar possíveis utilizações de matrizes esparsas para melhorar o desempenho nos cálculos de redes grandes.

REFERÊNCIAS

- ANJO, Luiz Fernando Resende dos Santos. **Modelo Hidráulico para Transitórios Lentos em Conduto Forçado**. Campinas, p. 154. 2008. Tese (Doutorado em Engenharia) - Faculdade de Engenharia Civil, Arquitetura e Urbanismo, Universidade de Campinas, Campinas, 2008.
- BORGES, Geovany Araújo. GMATRIX: Uma biblioteca matricial para C/C++. **Laboratório de automação e robótica**, 2005. Disponível em: <<http://lara.unb.br/~gaborges/recursos/programacao/index.htm>>. Acesso em: 15 jul. 2014.
- BRUNETTI, Franco. **Mecânica dos fluidos**. 2ª. ed. São Paulo: Pearson Prentice Hall, 2008. 420 p.
- CABRERA, Enrique Marcet; GARCIA-SERRA, Jorge; IGLESIAS, P. L. **Modelización de redes de distribución de água. Desde el régimen permanente hasta el golpe de aríete, “Mejora del rendimiento y de la fiabilidad em sistemas de distribución de água”**. Espanha: Cabrera, E.; Vela, A. F.; UPV, 1994. 536 p.
- CANTÙ, Marco. **Dominando o Delphi 7: a bíblia**. São Paulo: Pearson Makron Books, 2003. 801 p.
- EPA - United States Environmental Protection Agency. **EPANET - Software That Models the Hydraulic and Water Quality Behavior of Water Distribution Piping Systems**, 2013. Disponível em: <<http://www.epa.gov/nrmrl/wswrd/dw/epanet.html>>. Acesso em: 03 nov. 2013.
- FIALHO, Arivelto Bustamante. **Automação hidráulica: projetos, dimensionamento e análise de circuitos**. 2ª. ed. São Paulo: Érica, 2003. 284 p.
- GOMES, Heber Pimentel (Org.). **Sistemas de Saneamento - Eficiência Energética**. 1ª Edição. ed. João Pessoa: Universitária – UFPB, 2010. 366 p.
- HELLER, Léo; PÁDUA, Valter Lúcio de (Org.). **Abastecimento de água para consumo humano**. 2ª. ed. Belo Horizonte: UFMG, 2010. 872 p.
- HORSTMANN, Cay. **Conceitos de Computação com o Essencial de C++**. 3ª. ed. São Paulo: Bookman, 2005. 711 p.
- KHOSROW-POUR, Mehdi. **Dictionary of Information Science and Technology**. Hershey: IGI Global, v. 1, 2006. 1016 p.
- LEÃO, Marcelo. **Borland Delphi 7 Curso Completo**. Rio de Janeiro: Axcel Books, 2003. 1399 p.
- LUVIZOTTO JÚNIOR, Edevar. **Controle operacional de redes de abastecimento de água auxiliado por computador**. São Paulo, p. 143. 1995. Tese (Doutorado em Engenharia) - Escola Politécnica, Universidade de São Paulo, São Paulo, 1995.

MACKE, Steffen. Epanet.de Hydraulic network analysis. **EPANET and C: Build the EPANET toolkit DLL**, 2014. Disponível em: <<http://epanet.de/developer/index.html>>. Acesso em: 20 fev. 2014.

MICROSOFT Developer Network. **DLLs**, 2014a. Disponível em: <[http://msdn.microsoft.com/library/1ez7dh12\(v=vs.90\).aspx](http://msdn.microsoft.com/library/1ez7dh12(v=vs.90).aspx)>. Acesso em: 08 mar. 2014a.

MICROSOFT Developer Network. **How to: Debug Native DLLs**, 2014b. Disponível em: <<http://msdn.microsoft.com/library/c91k1xcf.aspx>>. Acesso em: 08 mar. 2014b.

NETTO, José Martiniano de Azevedo et al. **Manual de hidráulica**. 8ª. ed. São Paulo: Edgard Blucher, 1998. 669 p.

OLAIA, Ana Isabel Santos. **Gestão de Sistemas de Abastecimento de Água através de Modelação Hidráulica**. Lisboa, p. 120. 2012. Dissertação (Mestrado em Engenharia) - Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Lisboa, 2012.

PINTO, António Augusto Sampaio. **Gestão e Monitorização do Sistema Público de Distribuição de Água de Fafe - Cumieira**. Vila Real. 2009. Dissertação (Mestrado em Engenharia) - Universidade de Trás-os-Montes e Alto Douro, Vila Real, 2009.

PORTO, Rodrigo de Melo. **Hidráulica básica**. 4ª. ed. São Carlos: EESC-USP, 2006. 519 p.

ROSSMAN, Lewis A. EPANET 2 Users Manual. **EPANET**, 2000. Disponível em: <<http://nepis.epa.gov/Adobe/PDF/P1007WWU.pdf>>. Acesso em: 20 fev. 2014.

SABESP. Tratamento água. Disponível em: <http://site.sabesp.com.br/uploads/file/flash/tratamento_agua.swf>. Acesso em: 15 fev. 2014.

SCHILDT, Herbert. **C, completo e total**. Tradução de Roberto Carlos Mayer. 3ª. ed. São Paulo: Pearson Makron Books, 1997. 827 p.

TIPLER, Paul Allen; MOSCA, Gene. **Física para cientistas e engenheiros**. Tradução de Paulo Machado Mors. 6ª. ed. Rio de Janeiro: LTC, v. 1, 2012. 759 p.

TSUTIYA, Milton Tomoyuki. **Abastecimento de água**. 3ª. ed. São Paulo: Departamento de Engenharia Hidráulica e Sanitária da Escola Politécnica da Universidade de São Paulo, 2006. 643 p.

WARNER, Scott L.; GOLDSMAN, Paul. **Delphi 2 em exemplos**. Tradução de Edmilson Kazuyoshi Miyasaky. São Paulo: Makron Books, 1996. 551 p.

ANEXO A – AUTORIZAÇÃO PARA USO DO GMATRIX

De: Geovany A. Borges <gaborges@unb.br>
 Enviado em: terça-feira, 12 de agosto de 2014 00:20
 Para: Gustavo
 Assunto: Re: Autorização para uso GMATRIX

Caro Gustavo,

Claro, pode ficar à vontade.

Cordialmente,

Prof. Geovany Araujo Borges, DSc.

 Laboratório de Automação e Robótica (LARA) Departamento de Engenharia Elétrica (ENE) Campus
 Universitário Darcy Ribeiro Universidade de Brasília (UnB) Caixa Postal 4386 CEP 70919-970
 Brasília, DF - Brasil Tel. (office): (+55) 61 3107 5570
 Tel. (lab): (+55) 61 3107 1040
 Fax.: (+55) 61 3107-5590
 Homepage: <http://lara.unb.br/~gaborges/>

Em 11/08/2014 14:08, Gustavo escreveu:

> Geovany,
 >
 > Sou professor do Instituto Federal do Triângulo Mineiro – Campus
 > Uberaba e estou fazendo mestrado em Inovação Tecnológica na UFTM –
 > Universidade Federal do Triângulo Mineiro.
 >
 > Estou implementando no software EPANET um método de cálculo novo
 > denominado MDIR (Modelo Dinâmico Inercial Rígido).
 >
 > Estudei a biblioteca GMATRIX e fiquei muito satisfeito com os resultados.
 >
 > Gostaria de solicitar permissão para utilizar a biblioteca GMATRIX em
 > meu trabalho de mestrado.
 >
 > Atenciosamente.
 >
 > Prof. Gustavo Marino Botta
 > IFTM – Campus Uberaba
 >