

Flávio José Aguiar Soares

# Metodologia de Projeto de Bomba Dinâmica por Lâmina Ondulante

Tese apresentada à Escola Politécnica  
da Universidade de São Paulo para  
obtenção do Título de Doutor em Enge-  
nharia.

São Paulo  
Março de 2008

# **Livros Grátis**

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

Flávio José Aguiar Soares

# Metodologia de Projeto de Bomba Dinâmica por Lâmina Ondulante

Tese apresentada à Escola Politécnica  
da Universidade de São Paulo para  
obtenção do Título de Doutor em Enge-  
nharia.

área de concentração:  
Controle e Automação Mecânica

Orientador:  
Prof. Dr. Raul Gonzalez Lima

São Paulo  
Março de 2008

## Ficha Catalográfica

Aguiar Soares, Flávio José

Metodologia de Projeto de Bomba Dinâmica por Lâmina Ondulante. São Paulo, Março de 2008. 102 p.

Tese (Doutorado) — Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecânica

1. bioengenharia. 2. controle de estruturas flexíveis. 3. bombas dinâmicas I. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecânica . II. Título.

## Dedicatória

A minha amada esposa e filhas que junto comigo cruzaram este oceano

† In Memoriam  
Pe. Augusto Gianola.  
1930 - 1990  
Misionário do Pime no Amazonas.

“Senhor, Te adoro, mesmo se não sei o que isso significa.

Te agradeço, mesmo que só com palavras.

Te peço perdão, mesmo sem uma única lágrima.

Te ofereço tudo, mesmo se não possuo nada.

Desejo Te Amar, mesmo se não sou absolutamente capaz.”

# Agradecimentos

Verdadeiramente impossível tecer justo agradecimento a todos os que me ajudaram ao longo deste trabalho. Entretanto, registro algumas das pessoas as quais agradeço pela contribuição direta e indireta, pelo incentivo, apoio e auxílio.

Ao meu orientador Prof. Dr. Raul Gonzalez Lima, pela orientação precisa, dedicação incansável, apoio e estímulo inestimáveis.

A Ana Cláudia, companheira de todas as horas, esposa e mãe amorosa. As filhas Ana Débora e Ana Flávia, que juntas me deram um lar e o cultivaram num ambiente de paciência e carinho ao longo destes anos de trabalho.

Ao colega Elvys Mercato pela companhia, amizade, solidariedade, contribuição e apoio. Ao também colega Juan Cutipa pela companhia e amizade.

Aos Amigos Walter Britto e Oswaldo Dias, ao lado dos quais as horas tem um outro significado.

Aos professores do programa de pós-graduação, aos técnicos e funcionários administrativos da Escola Politécnica, com os quais convivi estes anos de experiência enriquecedora. Aos quais sou grato, pela amizade, pelas grandes e pequenas experiências do dia a dia.

Minha eterna gratidão ao professor Raimundo Vicente Gimenez, diretor do Cefet-AM, pelos anos que me permitiu vivenciar a Escola Politécnica.

À Fundação de Amparo a Pesquisa do Estado do Amazonas, pelo apoio financeiro concedido a esta pesquisa, sem o qual ela não teria se realizado.

# Resumo

Neste trabalho é apresentada uma metodologia de projeto de bomba dinâmica por lâmina ondulante. Um algoritmo permite determinar variáveis de projeto da bomba que maximizem a eficiência energética. A bomba consiste em um canal de seção transversal quadrada, dentro do qual há uma lâmina flexível, articulada nas extremidades. Impõe-se deformação transversal nessa lâmina. A deformação deve ter forma senoidal que se propaga ao longo da lâmina. A amplitude e a frequência de oscilação são controladas. Os campos de pressão e velocidade no canal foram estimados por simulação numérica, uma vez que não é possível obter uma solução analítica. O Método dos Elementos Finitos foi empregado para resolver a equação de Navier-Stokes. O movimento da lâmina foi representado através do conceito de fronteiras virtuais móveis. Uma vez obtidos os campos de pressão e velocidade, calcula-se a eficiência energética para avaliar quais variáveis de projeto geram maior eficiência energética. Doze atuadores eletromagnéticos aplicam forças à lâmina. O estado da lâmina é observado por cinco sensores de posição capacitivos. O comprimento da lâmina, a largura da lâmina, amplitude das oscilações, a frequência das oscilações, altura do canal e largura do canal são variáveis de projeto. Uma tabela que relaciona variáveis de projeto e eficiência mecânica foi calculada. Os resultados de simulação numérica indicam que é possível obter eficiência energética média da ordem de 20%. Resultados experimentais mostram o escoamento na saída da bomba. Este escoamento ocorre por golfadas em lados alternados da lâmina ondulatória.



# Abstract

In the present work a design methodology of undulatory pumps is proposed. An algorithm determines the design variables which maximize energy efficiency. The pump consists of a channel, with square transversal area, and a flexible blade hinged at its extremities. A sinusoidal transversal deformation is imposed by electromagnetic forces. The transversal deformation has sinusoidal shape and propagates along the channel, it resembles the movement of anguilliform fishes. The amplitude and frequency of oscillations are controlled. The pressure and velocity fields in the channel are computed by numerical simulations since analytical solutions are not possible. The Finite Elements Method is used to solve the Navier-Stokes equations. The blade movement is represented through the concept of virtual moving boundaries. Once the pressure and velocity fields are computed, the mechanical efficiency of the pump is computed to determine which set of design parameters generates higher energy efficiency. Twelve electromagnetic actuators apply forces on the blade. The state of the blade is observed by five capacitive position sensors. Amplitude of oscillations and frequency of oscillations are design variables. A map from design variables space to efficiency was computed. The results from numerical simulations suggest that it is possible to generate energy efficiency close to 20%. Experimental results show the flow at the outlet of the pump. The flow is more intense on alternate sides of the undulatory beam.

# Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

Convenções e Lista de Símbolos

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Concepção da Bomba . . . . .	3
1.2	Aspectos gerais . . . . .	3
1.3	Objetivos . . . . .	4
<b>2</b>	<b>Metodologia</b>	<b>5</b>
2.1	Desenvolvimento de um modelo numérico . . . . .	5
2.1.1	Movimento da lâmina . . . . .	5
2.1.2	Impondo movimento à fronteira móvel . . . . .	6
2.1.3	Volume de controle e condições de contorno . . . . .	7
2.2	Equações governantes . . . . .	8
2.3	Processador de elementos finitos para fluido dinâmica . . . . .	9
2.3.1	Especificidade do processador de elementos finitos . . . . .	10
2.3.2	Índice de desempenho . . . . .	11
2.4	Desenvolvimento do modelo experimental . . . . .	15
2.4.1	Atuadores eletromagnéticos . . . . .	16
2.4.2	Sensor de posição . . . . .	17
2.4.3	A lâmina propulsora . . . . .	18

2.4.4	Estratégia de controle do movimento da lâmina . . . . .	19
<b>3</b>	<b>Resultados</b>	<b>24</b>
3.1	Resultados do modelo numérico . . . . .	24
3.1.1	Visualização dos resultados . . . . .	26
3.1.2	Resposta em função da frequência . . . . .	30
3.2	Resultados experimentais do protótipo . . . . .	35
3.2.1	Medição da posição da lâmina . . . . .	35
3.2.2	Medição da força de atração eletromagnética . . . . .	37
3.2.3	Descrição da bomba ondulatória experimental . . . . .	42
<b>4</b>	<b>Comentários Finais</b>	<b>48</b>
<b>5</b>	<b>Conclusões</b>	<b>50</b>
<b>6</b>	<b>APÊNDICE A - Fronteira móvel</b>	<b>52</b>
6.1	Arquivos de definições da fronteira móvel . . . . .	52
6.1.1	Construção da malha . . . . .	52
6.1.2	Definição das condições de contorno . . . . .	52
6.1.3	Parâmetros da fronteira móvel . . . . .	53
6.1.4	Parametrizações globais . . . . .	53
6.2	Região de alcance da fronteira móvel . . . . .	53
6.3	Nós pertencentes à fronteira móvel . . . . .	54
6.4	Nós internos à fronteira móvel . . . . .	55
6.5	Componentes da velocidade da fronteira móvel . . . . .	56
<b>7</b>	<b>APÊNDICE B - Algoritmos em linguagem C</b>	<b>57</b>
7.1	Cálculo do índice de desempenho . . . . .	57
7.2	Algoritmo de controle de acionamento da bomba . . . . .	70
<b>8</b>	<b>APÊNDICE C - Algoritmos em FORTRAN77</b>	<b>86</b>

8.1	KL1.0 . . . . .	86
8.2	parq2d.f . . . . .	88
8.3	indat2d.f . . . . .	89
8.4	bndry.f . . . . .	95
8.5	cc2d.dat . . . . .	98
8.5.1	Obsevações sobre o uso do Featflow . . . . .	100

<b>Referências</b>	<b>101</b>
--------------------	------------

# Lista de Figuras

2.1	O movimento da lâmina é uma função senoidal modulada. . . . .	6
2.2	Formato do canal de escoamento correspondente ao volume de controle. . . . .	7
2.3	Vínculos de fixação da lâmina ao canal. . . . .	7
2.4	Forças agindo sobre um elemento da lâmina. . . . .	12
2.5	Nós empregados no cálculo da pressão no <i>i-ésimo</i> elemento de lâmina.	15
2.6	Sensor de posição diferencial capacitivo teórico. . . . .	17
2.7	Esquema eletrônico do circuito de tratamento do sinal dos sensores diferenciais capacitivos. . . . .	19
2.8	Representação da lâmina, sensores e atuadores. . . . .	20
2.9	Modelo dinâmico de elementos finitos da lâmina. . . . .	20
2.10	Seis primeiros modos naturais de vibração para o modelo MEF da lâmina. . . . .	23
2.11	Gráfico dos pólos de malha aberta. . . . .	23
3.1	Amplitudes da senoide para uma mesma frequência e uma mesma fase. . . . .	25
3.2	Deslocamento longitudinal da deformação transversal no tempo. . . . .	25
3.3	O número de Reynolds em função da amplitude de deformação transversal da lâmina. . . . .	25
3.4	A eficiência energética em função da amplitude de deslocamento, do número de Reynolds e da frequência. . . . .	26
3.5	Perfil do escoamento da velocidade na seção de entrada ainda não desenvolvido. . . . .	27
3.6	Perfil do escoamento da velocidade na seção de saída ainda não desenvolvido. . . . .	27

3.7	Campo de pressões para frequência de 5 Hz, de 0,05 a 0,010s, de cima para baixo. . . . .	28
3.8	Campo de velocidade em $m/s$ na direção $x$ para frequência de 15Hz, fluxo da esquerda para a direita. . . . .	28
3.9	Comportamento pulsátil da eficiência mecânica, amplitude 0.0025m, frequências de 1 a 20Hz. . . . .	29
3.10	Velocidade média na seção de entrada versus tempo para diversos valores de frequência e amplitude. . . . .	29
3.11	Ganho de pressão média versus tempo para valores de frequência de 1 a 15Hz. . . . .	30
3.12	Ganho de pressão média versus tempo na frequência de 20Hz para valores de amplitude de deslocamento transversal de 0.0010 a 0.0025 m. . . . .	31
3.13	Vazão média versus tempo para frequências de 1 a 15Hz. . . . .	31
3.14	Vazão média versus tempo para diversas amplitudes mantendo a frequência em 20Hz. . . . .	32
3.15	Eficiência mecânica média versus tempo para frequências de 1 a 15Hz. . . . .	32
3.16	Eficiência mecânica média versus tempo para diversas amplitudes mantendo a frequência em 20Hz. . . . .	33
3.17	Eficiência média em função da frequência quando a extremidade da lâmina próxima à saída do canal pode apenas transladar longitudinalmente. . . . .	33
3.18	Ganho de pressão médio versus frequência para diversas amplitudes de deslocamento transversal. . . . .	34
3.19	Eficiência média em função da frequência quando a extremidade da lâmina próxima à saída do canal está livre. . . . .	34
3.20	Bomba experimental com 12 atuadores eletromagnéticos ao longo do canal. . . . .	35
3.21	Vista explodida dos componentes da Bomba experimental. . . . .	35
3.22	Potencial elétrico teórico de um sensor capacitivo de posição. . . . .	36
3.23	Sinal de tensão obtido para cada sensor de posição. . . . .	37

3.24	Sinais de tensão dos sensores invertidos e interpolados. . . . .	37
3.25	Comportamento do sinal de saída de um sensor diferencial capacitivo.	37
3.26	Aspecto final do eletromagneto. . . . .	38
3.27	Dispositivo de medição da força eletromagnética. . . . .	39
3.28	Vista explodida do dispositivo de medição da força magnética. . .	39
3.29	Curva de calibração da célula de carga. . . . .	40
3.30	Curva de força eletromagnética versus distância da lâmina ao eletromagneto. . . . .	40
3.31	Força de atração magnética versus espessura da lâmina. . . . .	41
3.32	Força de atração magnética em função da distância e da corrente.	41
3.33	Força de atração versus corrente para diversos valores de distância, as menores distâncias geram forças mais elevadas. . . . .	42
3.34	Força de atração versus distância para diversos valores de corrente, os maiores valores de corrente geram forças mais elevadas. . . . .	42
3.35	Esquema de integração do <i>hardware</i> de controle. . . . .	43
3.36	Circuito da fonte de corrente de acionamento dos eletromagnetos.	44
3.37	Comportamento linear do circuito de acionamento dos eletromagnetos. . . . .	45
3.38	Vista superior do protótipo experimental. . . . .	46
3.39	Vista superior da seção de descarga do protótipo experimental. . .	46
3.40	Escoamento pulsátil na seção de descarga da bomba. . . . .	47
6.1	O retângulo representa a fronteira fixa e a região escura contém os nós que podem pertencer a Fronteira Móvel. . . . .	54
6.2	Determinação dos nós pertencentes à fronteira móvel. . . . .	54

# Lista de Tabelas

2.1	Permissividade elétrica ou constante dielétrica. . . . .	18
3.1	Parâmetros de projeto . . . . .	24



# Lista de Abreviaturas

**2D** Duas dimensões

**3D** Três dimensões

**BE** Backward Euler

**CN** Crank-Nikolson

**EF** Elementos Finitos

**FR** Fractional-step- $\theta$ -scheme

**CN** Fronteira Artificial Móvel

**PC** Computador Pessoal

**AUV** Veículos Submersíveis Autônomo

**CFD** Escoamento Computacional Fluido Dinâmico

**EDP** Equação Diferencial Parcial

**GMV** General Mesh Viewer

**MEF** Método dos Elementos Finitos

# Convenções e Lista de Símbolos

Na notação das fórmulas, as seguintes convenções foram utilizadas:

- letras maiúsculas em negrito são matrizes, exemplos: **A**, **B**, **C**;
- letras minúsculas sobrescritas com uma seta são vetores, exemplos:  $\vec{x}$ ,  $\vec{y}$ ,  $\vec{z}$ ;

Os seguintes símbolos serão utilizados:

símbolo	descrição	página
$\omega$	deformação da lâmina	5
$x$	posição ao longo da lâmina	5
$t$	tempo contínuo	5
$A_{mp}$	amplitude de deslocamento da onda sobre a lâmina	6
$L$	comprimento da lâmina	6
$a$	parâmetro de definição do comprimento de onda sobre a lâmina	6
$\phi$	velocidade angular da onda sobre a lâmina	6
$p$	pressão normalizada	8
$\vec{u}$	vetor velocidade do escoamento	8
$\vec{\ddot{u}}$	vetor aceleração do escoamento	8
$\nu$	viscosidade cinemática	8
$\vec{f}$	vetor das força de campo	8
$P_{ef}$	pressão efetiva	8
$\rho$	massa específica do fluido	8
$P_{ef}$	pressão efetiva	9
$\Gamma$	coeficiente de difusão	10
$\eta$	rendimento energético	11
$T_{out}$	trabalho que sai da bomba	11
$T_{in}$	trabalho que entra na bomba	11
$T_{fm}$	trabalho realizado pela fronteira artificial móvel	11
$p_{in}$	pressão normalizada na entrada da bomba	11
$A_{in}$	área da seção de entrada da bomba	11

*continua...*

símbolo	descrição	página
$\Delta x_{in}$	deslocamento do elemento de fluido na entrada da bomba	11
$\Delta t$	incremento de tempo discreto	11
$v_{in}$	velocidade do elemento de fluido na entrada da bomba	11
$p_{out}$	pressão normalizada na saída da bomba	12
$A_{out}$	área da seção de saída da bomba	12
$\Delta x_{out}$	deslocamento do elemento de fluido saída da bomba	12
$v_{out}$	velocidade do elemento de fluido na saída da bomba	12
$F_i$	força resultante sobre o i-ésimo elemento da lâmina	12
$\Delta y$	deslocamento produzido pela força $F_i$	12
$f_c^-$	força cortante negativa sobre um i-ésimo elemento da lâmina	12
$f_c^+$	força cortante positiva sobre um i-ésimo elemento da lâmina	12
$P^-$	pressão média na face inferior de um i-ésimo elemento da lâmina	12
$P^+$	pressão média na face superior de um i-ésimo elemento da lâmina	12
$\ddot{\omega}$	aceleração produzida pela força $F_i$	12
$M$	momento de uma força	13
$E$	módulo de elasticidade	13
$I$	momento de inércia	13
$\Delta m$	elemento de massa da lâmina	15
$\Delta l$	espessura da lâmina	15
$\Delta z$	comprimento do elemento da lâmina	15
$\rho_{lamina}$	densidade de massa da lâmina	15
$f_{mag}$	força de atração eletromagnética	16
$\mu_0$	permeabilidade magnética no vácuo	16
$n_e$	número de espiras	16
$i$	corrente elétrica	16
$l$	dimensão lateral do núcleo	16
$d$	dimensão lateral complementar do núcleo	16
$y$	distância entre o núcleo e a lâmina	16
$\mu_{bomba}$	permeabilidade magnética entre as bobinas e a lâmina	16
$C$	capacitância	17

*continua...*

símbolo	descrição	página
$\epsilon_0$	constante dielétrica no vácuo	17
$\epsilon_r$	constante dielétrica relativa	17
$S$	área das placas capacitivas	17
$\epsilon_w$	constante dielétrica da água destilada	18
<b>M</b>	matriz de massa global	20
<b>K</b>	matriz de rigidez global	20
$F_n$	vetor das forças nodais	20
$\chi$	vetor de estados	21
<b>A</b>	matriz de estado	21
<b>B</b>	matriz de entrada	21
<b>C</b>	matriz de saída	21
$u$	sinal de saída do estado	21
$y_{ref}$	estado de referência	22
$y_{med}$	estado medido	22
<b>K<sub>g</sub></b>	matriz de ganho	22
$Re$	número de Reynolds	25
$ppp$	distância entre dois nós sobre a malha	53
$XX$	ordenada de um nó qualquer da malha	53
$YY$	abscissa de um nó qualquer da malha	53
$X_{m1}$	ordenada da origem do sistema de posição da lâmina	53
$Y_{m1}$	abscissa da origem do sistema de posição da lâmina	53
$\pi_2$	variável auxiliar	55
$\omega_f$	equação de deslocamento para a deformação da lâmina	55
$n$	número de intervalos ao longo de uma simulação	55
$\pi_{2d}$	variável auxiliar	55
$\pi_{1d}$	variável auxiliar	55
$d_1$	variável auxiliar para avaliar espessura da fronteira artificial móvel	55
$v_x$	velocidade de escoamento na direção $\vec{x}$	56
$v_y$	velocidade de escoamento na direção $\vec{y}$	56

# 1 Introdução

Neste trabalho é proposta uma metodologia de projeto de uma bomba que utiliza uma lâmina flexível como impelidor do fluido, esta lâmina deve se deformar imitando o movimento de uma moréia, que se assemelha a uma função seno que se desloca ao longo de uma lâmina.

Em 1936 o zoologista James Gray enunciou o argumento que ficou conhecido como paradoxo de Gray, Triantafyllou e Triantafyllou (1995). Ao estudar a potência propulsiva de golfinhos, Gray percebeu que a capacidade muscular era sete vezes inferior aos requeridos para o desempenho apresentado. Apenas em 1996, Triantafyllou (1996), apresentou um modelo matemático que respondia ao paradoxo, demonstrando que o movimento da cauda dos peixes geram vórtices alternados e com sentidos de rotação opostos, o que permite a redução do arrasto sobre os corpos dos peixes. Alguns trabalhos, Webb (1983), sobre a cinemática de peixes também serviram de base para diversas pesquisas posteriores, Walker (2004).

Este método de propulsão, empregado em peixes robóticos, permitiu que obtivessem rendimentos da ordem de 86%, muito superiores aos valores encontrados para os propulsores convencionais empregados para deslocar embarcações, que possuem rendimentos máximos em torno de 40%.

Este desempenho encorajou o estudo de movimentos de diversos peixes que pudessem ser aplicados no desenvolvimento de novas formas de acionamento de propulsores para veículos submersíveis autônomos (AUV) e para bombas. Os autores Nakashima e Ono (2000) apresentam simulação numérica para um mecanismo com três elementos articulados que alcançou eficiência teórica de 65% em condições controladas.

Exemplos de robôs biomiméticos em forma de peixe foram desenvolvidos, Yu (2004) Terada e Yamamoto (2004). Outros, Liu e Bose (1997), apresentam um estudo sobre o desempenho de propulsores oscilantes a partir da morfologia de baleias, que também podem apresentar rendimentos acima de 80%. Um detalhado

estudo biológico da morfologia de barbatanas de peixes e ensaios hidrodinâmicos para aplicação em AUVs pode ser visto em Lauder G. V. (2004). Walker (2002) apresenta estudo sobre a funcionalidade morfológica de alguns animais e o número de Reynolds, sugerindo que existem faixas do número de Reynolds em que um tipo de propulsão é mais eficaz que outras. O número de Reynolds define se é mais eficiente nadar, remar ou voar. Walker (2004) apresenta outro artigo, direcionado para o projeto de AUVs baseado na morfologia de peixes, o foco está no desempenho hidrodinâmico. Uma revisão dos trabalhos experimentais com propulsores oscilantes baseados na morfologia de peixes marinhos pode ser vista em Triantafyllou M. S. (2004).

O mesmo princípio tem sido empregado por alguns autores no desenvolvimento de bombas com lâminas oscilantes Laser e G. (2003), Techet (1999), Triantafyllou e Triantafyllou (1995). De um lado existem as bombas que imitam peixes *ostraciformes*, nos quais os movimentos da lâmina é denominado *oscilatório*, com propulsão definida por um movimento de calda, Nakasone P. H. (2005). Do outro lado existem bombas que imitam peixes *anguiliformes*, com a propulsão definida por um movimento de corpo inteiro, empregado por moréias ou cobras d'água., nos quais o movimento da lâmina é denominado *ondulatório*. A metodologia de projeto desenvolvida neste trabalho se refere a movimentos ondulatórios da lâmina.

Um estudo que utiliza princípios similares ao presente trabalho, quanto ao modo físico de acionamento do elemento impelidor, foi apresentado por Yamahata C. (2005), nele está descrita uma bomba cujo propulsor é uma esfera acionada por eletromagnetos, operando numa faixa de frequência entre 20 e 30Hz.

A Universidade de Osaka, Robot... (2008, Acesso em 20 de Fevereiro de 2008), apresenta um vídeo de um robô submersível que se movimenta mimetizando o movimento de uma lula. Nele são empregados duas membranas laterais. Sobre a aresta externa de cada membrana se impõe deformação segundo uma senoide progressiva. A medida que a senoide se propaga, o robô se desloca em sentido contrário. Este trabalho utiliza o mesmo princípio. Mantendo uma extremidade de uma lâmina flexível fixa a um canal, se impõe um deslocamento em forma de senoide progressiva sobre a lâmina flexível e espera-se que o fluido se desloque no sentido oposto.

## 1.1 Concepção da Bomba

Peixes *anguiliformes* apresentam movimento suaves e lentos. Espera-se que as tensões de cisalhamento e as taxas de deformação também sejam baixas. Uma concepção de bombeamento que siga este modelo pode ser interessante em fluidos como o sangue, onde a taxa de hemólise<sup>1</sup> é função das tensões de cisalhamento sobre o fluido. Este trabalho é uma tentativa de otimizar este tipo de bomba.

Foi empregada uma concepção de projeto que procurou representar o movimento da moréia, empregando o mínimo de elementos possíveis.

O propulsor escolhido, que deve representar movimento de moréia foi uma lâmina flexível. A lâmina foi fixada dentro de um canal de seção quadrada. As fixações foram posicionadas na metade da largura do canal. Foram avaliadas lâminas com espessuras abaixo de 0,30mm de modo a reduzir força para deformar a viga

O movimento *anguiliforme* foi modelado por uma senoide progressiva, que se propaga ao longo da lâmina. Tal senoide deve ser imposta à lâmina utilizando-se duas fileiras de atuadores eletromagnéticos dispostos uma em frente a outra, de cada lado do canal.

Para impor movimento à lâmina foram escolhidos atuadores eletromagnéticos, de modo a se poder mover a lâmina utilizando forças de campo e não de contato, com isso se evitou que eixos ou outros dispositivos fossem introduzidos no canal, criando pontos de oposição ao fluxo, aumentando a perda de carga e reduzindo a eficiência de bombeamento.

Para detectar a posição que alguns pontos da lâmina assumem ao longo do movimento e permitir a realimentação da malha de controle, foram escolhidos sensores diferenciais capacitivos, que fisicamente correspondem a dois eletrodos colocados em faces opostas do canal com a lâmina variando de posição entre eles. A espessura das placas dos eletrodos são da mesma ordem de grandeza da lâmina, de modo a gerarem um mínimo de interferência no fluxo do fluido de trabalho.

## 1.2 Aspectos gerais

Procura-se otimizar a bomba, adotando-se o rendimento energético como índice de desempenho. Isto é feito variando-se parâmetros do modelo através de simulações

---

<sup>1</sup>morte de hemácias

numéricas em 2D. Para que o rendimento energético possa ser calculado é preciso conhecer os campos de pressão e velocidade do escoamento ao longo do tempo. Os campos de pressão e velocidade foram calculados através da solução numérica da equação de Navier-Stokes. As simulações foram realizadas empregando-se o programa de escoamento computacional fluido dinâmico (CFD) Featflow1.3RC3, disponível em Featflow... (2007, Acesso em 12 de maio de 2007)<sup>2</sup>.

Este programa implementa o MEF e permite a utilização de fronteiras artificiais móveis, o que corresponde a uma região do domínio que pode assumir velocidade e pressão impostas pelo usuário. A medida que a fronteira artificial se move, os campos de velocidade e de pressão são alterados, impondo o deslocamento do fluido. A fronteira móvel é um domínio móvel dentro de um domínio fixo. O domínio fixo é representado por uma malha de elementos finitos. Segundo Turek (1998) elas podem ser implementadas sem que a malha sofra deformação.

### 1.3 **Objetivos**

Este trabalho visa desenvolver uma metodologia de projeto de bombas dinâmicas por lâmina flexível. O propulsor da bomba é uma lâmina em movimento ondulatório, acionado por forças de campo magnético variável. A metodologia de projeto envolve a simulação do escoamento e o cálculo da eficiência energética de uma particular escolha de variáveis de projeto. O método deverá permitir a escolha de variáveis de projeto que resultem em vazão e ganho de pressão especificadas. A forma construtiva da bomba também deve ser apresentada.

---

<sup>2</sup>Trata-se de uma versão beta de distribuição gratuita e de código aberto, escrito em FORTRAN77



## 2 Metodologia

Para desenvolver um método de projeto de bombas ondulatórias serão desenvolvidos um *modelo numérico* de bomba ondulatória e um *modelo experimental* de uma bomba ondulatória para avaliar a validade do conceito empregado. Desta forma, este trabalho deverá ter as seguintes etapas:

1. Desenvolvimento de um modelo numérico;
2. Desenvolvimento de um modelo experimental;
3. Cálculo do campo de velocidade, do campo de pressão, de eficiência energética, do ganho de pressão e de vazão para um conjunto de parâmetros de projeto;
4. Construção de um protótipo para a avaliar a viabilidade do conceito.

### 2.1 Desenvolvimento de um modelo numérico

#### 2.1.1 Movimento da lâmina

A lâmina flexível tem uma extremidade articulada e a outra extremidade apoiada. A deflexão transversal  $\omega$  de cada elemento  $dx$ , ao longo da lâmina oscilante, varia espacial e temporalmente, conforme a equação (2.1).

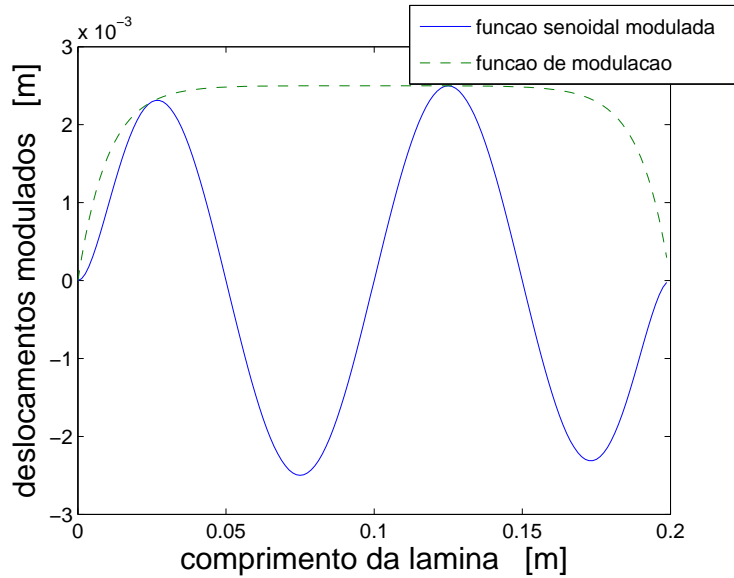
$$\omega = \omega(x, t) \tag{2.1}$$

Para definir o deslocamento da lâmina que se aproxime do movimento dos peixes anguiliformes utiliza-se o produto de duas funções. A primeira função é uma senoide que se propaga ao longo do comprimento da lâmina, ela é modulada por duas exponenciais de forma que a amplitude seja nula na articulação e pequena no apoio simples, ver equação (2.2). A função resultante será denominada *função de deslocamento*, ver equação (2.3). Gráficos destas funções encontram-se sobrepostos na Figura (2.1).

$$\omega_x = A_{mp} (1 - \exp^{-x}) (1 - \exp^{x-L}) \quad (2.2)$$

$$\omega_f(x, t) = A_{mp} (1 - \exp^{-x}) (1 - \exp^{x-L}) \operatorname{sen} \left( \left( \frac{\pi}{a} \right) x - \phi t \right) \quad (2.3)$$

onde  $A_{mp}$  é a amplitude de oscilação,  $x$  é a posição ao longo da lâmina,  $L$  é o comprimento da lâmina,  $a$  define o comprimento de onda,  $\phi$  é a velocidade angular e  $t$  o tempo. O avanço espacial da onda é dado pelo avanço do termo  $\phi t$ .<sup>1</sup>



**Figura 2.1:** O movimento da lâmina é uma função senoidal modulada.

### 2.1.2 Impondo movimento à fronteira móvel

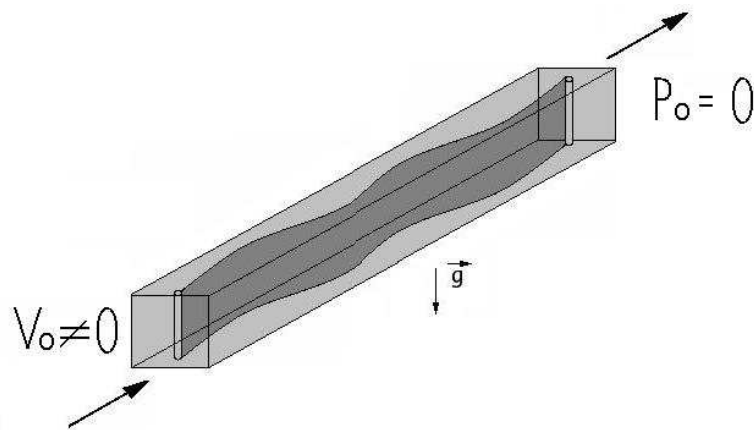
O canal de escoamento da bomba é o volume de controle sob o qual se calculam os campos de velocidade e pressão. Ele é modelado através de uma malha de elementos finitos e corresponde a um domínio fixo que delimitará a solução da equação de Navier-Stokes. Dentro deste domínio fixo se estabelece um domínio móvel, no qual um conjunto de nós tem suas variáveis associadas de velocidade e pressão impostas pelo usuário. Esse domínio móvel corresponde a uma fronteira fictícia interna ao volume de controle. Os nós da fronteira móvel pertencem à vizinhança da curva definida pela equação de deslocamento (2.3). A cada incremento temporal discreto  $\Delta t = 0.01s$ , determinam-se os nós da malha que pertencem à fronteira móvel e atualizam-se as condições de velocidade nestes nós.

<sup>1</sup>Amp = 0,0025 m, L = 0,20 m, a = 5,  $\phi = 15\text{Hz}$  e  $\Delta t = 0,01$  s.

O algoritmo reconhece quais nós pertencem ao domínio da fronteira móvel utilizando regras que definem sua localização. Estas regras são um conjunto de aproximações que restringem progressivamente a região da malha sobre a qual devem estar os nós pertencentes a fronteira móvel. O APÊNDICE A - *Fronteira móvel* apresenta o procedimento empregado para a formulação da fronteira móvel empregada neste trabalho.

### 2.1.3 Volume de controle e condições de contorno

O volume de controle corresponde a um duto de seção transversal quadrada<sup>2</sup>, cujos extremos são as seções de entrada e saída da bomba, de acordo com a Figura (2.2).



**Figura 2.2:** Formato do canal de escoamento correspondente ao volume de controle.

As duas extremidades da lâmina são fixas a dois eixos. O eixo junto a seção de entrada pode rotacionar sobre si. O eixo oposto também pode rotacionar sobre si e, além disso, pode transladar longitudinalmente. Essa combinação de vínculos permite que se estabeleça o movimento ondulatório sobre a lâmina. O sentido de deslocamento da onda sobre a lâmina é estabelecido pelo usuário. Ver Figura (2.3).



**Figura 2.3:** Vínculos de fixação da lâmina ao canal.

<sup>2</sup>área da seção =  $10^{-4} m^2$ , comprimento do canal =  $0,24 m$  e comprimento da lâmina =  $0,20 m$ , relação comprimento/largura =  $20/1$

As condições de contorno mantidas durante as simulações são, pressão nula na seção de saída, o perfil conhecido de velocidade na seção de entrada, o fluido tem velocidade tangencial nula na fronteira fixa e na fronteira móvel. O fluido de trabalho é viscoso, newtoniano e incompressível. Outras condições impostas são as dimensões da lâmina, a amplitude de deslocamento da lâmina e a velocidade de propagação da onda de deslocamento transversal da lâmina.

O procedimento de busca da solução de Navier-Stokes é não linear e iterativo. Valores de pressão a montante e a jusante são definidos na primeira iteração como nulos. A distribuição de velocidade no domínio na seção de entrada deve possuir valor não nulo, uma vez que, para que o algoritmo do Featflow possa convergir ele deve iniciar com um valor não nulo do campo de velocidades na entrada. Esta é uma condição necessária para a solução da equação de Navier-Stokes usando o Featflow.

A simulação ocorre em 2D e é preciso definir que face do volume de controle seria visualizada no pós-processamento. Escolheu-se a vista superior porque esta pode ser futuramente comparada com dados experimentais obtidos por métodos ópticos. A seção superior da montagem experimental é transparente. A aceleração da gravidade é ortogonal ao fluxo, e portanto não acelera o fluido, ver Figura (2.2).

## 2.2 Equações governantes

O escoamento de um fluido incompressível pode ser representado pelas equações de Navier-Stokes, mais especificamente, pela equação da quantidade de movimento (2.4) e pela equação da continuidade (2.5).

$$\vec{u} - \nu \Delta \vec{u} + \vec{u} \nabla \vec{u} + \nabla p = \vec{f} \quad (2.4)$$

$$\nabla \vec{u} = 0 \quad (2.5)$$

onde  $p$  é a pressão normalizada, conforme equação (2.6),  $\vec{u}$  é o vetor velocidade, e  $\vec{f}$  é o vetor das forças de campo, também normalizadas pela massa específica  $\rho$ .

A pressão normalizada é a razão da pressão efetiva  $P_{ef}$  e a massa específica  $\rho$ , de acordo com Turek (1998), sendo  $\rho$ , a massa específica, dada em  $\frac{kg}{m^3}$ . Esta

normalização faz parte do algoritmo FEATFLOW.

$$p = \frac{P_{ef}}{\rho} \quad (2.6)$$

## 2.3 Processador de elementos finitos para fluido dinâmica

Será empregado um algoritmo de solução da equação de Navier-Stokes que permite descrever condições de contorno em fronteiras que estão em movimento. O algoritmo Featflow permite representar fronteiras móveis através do conceito de fronteiras móveis virtuais, sem ter que movimentar as malhas do modelo. Adicionalmente, o Featflow é um processador baseado no Método dos Elementos Finitos de código aberto e com gerenciamento de memória adequado para ser utilizado em computadores pessoais. Todos os programas utilizados ao longo deste trabalho têm licença GNU e estão disponíveis na rede mundial de computadores.

Nesta seção descreve-se o processador de Elementos Finitos Featflow do ponto de vista do usuário. Uma vez adotadas as equações de Navier-Stokes para escoamentos incompressíveis, resta definir se o modelo do escoamento será bidimensional ou tridimensional e as condições de contorno nas quais a equação deve operar.

O Featflow começa por uma discretização temporal, o *Fractional-step -  $\theta$  - scheme*, da equação da quantidade de movimento, equação 2.4. O método numérico de integração temporal *Fractional-step -  $\theta$  - scheme* apresenta as vantagens dos métodos *Backward-Euler* e *Crank-Nicolson*, com o mesmo custo computacional, de acordo com Turek (1998).

O algoritmo dispõe de dois métodos para lidar com este problema, já discreto no tempo, através de um método tipo Galerkin ou através de um método de projeções. O método tipo Galerkin utiliza a equação de Oseen e o método baseado em projeções utiliza a equação de Burgers. A equação de Oseen (2.7), corresponde a equação de Navies-Stokes sem o termo convectivo. A equação de Burgers (2.8) é uma equação não linear parabólica do tipo convecção-difusão, onde o primeiro termo é convectivo e segundo termo difusivo. Possui soluções tabeladas e características comuns a Navier-Stokes.

$$\vec{u} - \nu \Delta \vec{u} + \nabla p = \vec{f} \quad (2.7)$$

$$\frac{\partial v}{\partial t} = -v \frac{\partial v}{\partial x} + \Gamma \frac{\partial^2 v}{\partial^2 x} \quad (2.8)$$

onde  $v$  é a velocidade,  $t$  o tempo,  $x$  a direção do deslocamento e  $\Gamma$  é o coeficiente de difusão.

### 2.3.1 Especificidade do processador de elementos finitos

O Featflow é um algoritmo escrito em FORTRAN77 e a entrada de dados é realizada por linhas do comando e alguns *scripts*. Para visualização do modelo e da solução utiliza-se o General Mesh Viewer (GMV), (GMV..., 2005, Acesso em 18 de julho de 2005). O Featflow não possui um pré-processador gráfico ou um CAD. Nele o usuário deve definir através da modificação de linhas de comando toda a parametrização do problema, define a densidade da malha, define as fronteiras do volume de controle, define as condições de contorno e define a forma, a frequência e a amplitude do deslocamento da fronteira móvel. Os arquivos solução *u.n.gmv* do *Featflow* apresentam os valores dos campos de pressão, velocidade e fluxo. Os campos de velocidade são descritos pelas direções  $\vec{u}$ ,  $\vec{v}$  e  $\vec{w}$ .

O algoritmo Featflow emprega uma hierarquia de malhas, procedimento denominado *multi-grid*, para encontrar a solução da equação de Navier-Stokes (2.4). Esta estratégia garante velocidade e estabilidade computacional. Utilizam-se elementos quadrilaterais com funções de forma bilineares, Turek (1998).

A definição do problema deve ser feita diretamente sobre arquivos que serão posteriormente compilados. Estes arquivos são apresentados no *APÊNDICE C - Algoritmos em FORTRAN77*. Estes arquivos são responsáveis pelas seguintes tarefas: geração da malha e definição do contorno do volume de controle - arquivos *KL1.0* e *parq2d.f*; definição das condições de contorno e fronteira móvel - arquivos *indat2d.f* e *bn dry.f*; definição da equação a ser resolvida e parâmetros do método numérico - arquivo *cc2d.dat*.

Para que a simulação aconteça de modo desejado é preciso que estes arquivos sejam coerentes entre si, eles se complementam. O algoritmo é pouco tolerante a falhas nos arquivos *parq2d.f* e *KL1.0*. Entretanto, entre todos os arquivos deve-se dar especial atenção ao arquivo de parametrização *cc2d.dat*. Pequenas alterações nas suas definições podem conduzir a soluções de problemas distintos.

Descrevem-se a seguir, de modo sucinto, os procedimentos necessários para realizar uma simulação. Documentação específica pode ser encontrada na página do programa na rede mundial de computadores, Featflow... (2007, Acesso em

12 de maio de 2007). Os algoritmos de manipulação de dados dos arquivos soluções do Featflow1.3RC3 que foram empregados para obter o valor do rendimento energético são apresentados no *APÊNDICE B - Algoritmos em linguagem C*.

### 2.3.2 Índice de desempenho

Conhecidos os campos de pressão e velocidade, pode-se calcular o trabalho realizado pelo fluido sobre o meio externo. Conhecido o movimento da lâmina e a pressão na superfície da lâmina pode-se calcular o trabalho realizado pelo meio externo sobre o fluido. A razão entre trabalho realizado pelo fluido na saída e o trabalho realizado sobre o fluido determina o rendimento da bomba.

A eficiência energética foi calculada considerando o trabalho realizado pela bomba  $T_{out}$ , o trabalho realizado sobre a bomba  $T_{in}$  na seção de entrada e o trabalho realizado sobre o fluido pela lâmina  $T_{fm}$ , veja equação (2.9).

$$\eta = \frac{T_{out}}{T_{in} + T_{fm}} \quad (2.9)$$

A força externa agindo em cada elemento do fluido é calculada como o produto da pressão sobre o elemento pela área do elemento. O termo  $T_{in}$ , ver equação (2.10), é o módulo da somatória do trabalho realizado sobre cada elemento de fluido na seção de entrada do volume de controle.

$$T_{in} = \sum_{i=1}^{33} p_{in}(i) A_{in} \Delta x(i)_{in} \quad (2.10)$$

onde  $i$  é o número de nós pertencentes à seção de entrada do volume de controle,  $p_{in}$  é a pressão na seção de entrada e  $\Delta x(i)$  é o deslocamento do elemento de fluido em certo intervalo de tempo. O deslocamento na direção  $x$  de cada elemento é o produto da velocidade na direção  $x$  deste elemento pelo intervalo de tempo entre simulações  $\Delta t$ , ver equação (2.11), ou seja, o deslocamento foi calculado a partir do perfil de velocidades obtido da seção de entrada do protótipo.

$$\Delta x(i)_{in} = v_{in}(i) \Delta t \quad (2.11)$$

O termo  $T_{out}$ , ver equação (2.12), é o módulo da somatória do trabalho realizado por cada elemento de fluido que sai da bomba. Sua determinação segue o procedimento empregado para determinar o termo  $T_{in}$ , entretanto, o desloca-

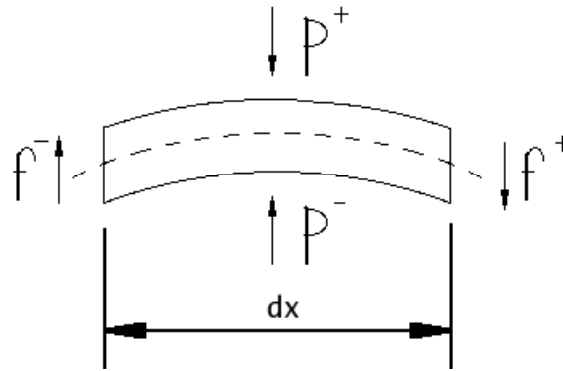
mento foi calculado a partir do perfil de velocidades obtido da seção de saída do volume de controle.

$$T_{out} = \sum_{i=1}^{33} p_{out}(i) A_{out} \Delta x(i)_{out} \quad (2.12)$$

onde,

$$\Delta x(i)_{out} = v_{out}(i) \Delta t. \quad (2.13)$$

O termo  $T_{fm}$ , ver equação (2.14), é a somatória do trabalho realizado por cada elemento da lâmina  $dx$ . A força resultante sobre cada elemento da lâmina é avaliada empregando-se a segunda lei de Newton, o que implica em considerar a diferença de pressão entre os lados da lâmina, as forças cortantes e as forças de inércia sobre a massa  $dm$  do elemento. Ver Figura (2.4).



**Figura 2.4:** Forças agindo sobre um elemento da lâmina.

$$T_{fm} = \sum_{q=1}^{160} F(i) \Delta y(i) \quad (2.14)$$

onde  $i$  é o número do elemento  $dx$  ao longo do comprimento da lâmina,  $F$  a força resultante, ver equação (2.15), e  $\Delta y$  o deslocamento do elemento.

O deslocamento  $\Delta y$  é obtido pela equação de deslocamento discretizada (2.18) para um intervalo de tempo  $\Delta t$  e um elemento da lâmina.

A força resultante pode ser decomposta nos seguintes termos:

$$\Delta m \ddot{w} = f_c^- + f_c^+ + \bar{P}^- - \bar{P}^+ + F \quad (2.15)$$



### 2.3.2.1 Esforço cortante

Foi aplicado o modelo de viga Euler-Bernoulli. Este modelo é válido quando:

- O comprimento da viga tem comprimento muito maior que as dimensões de sua seção transversal, e que deformação devida a seu próprio peso é desprezível.
- A seção da viga não muda quando é curvada, quando a espessura da viga é pequena comparado com o raio de curvatura, a seção transversal varia muito pouco.

Para as condições adotadas, nota-se que a lâmina tem comprimento 1000 vezes maior que sua espessura e seu peso estimado é de 0,032N. Considerando que o comprimento de onda é de aproximadamente 0,12m, a espessura da viga é 300 vezes menor que o raio de curvatura.

Verifica-se portanto que as condições de validade são cumpridas. Em decorrência do uso deste modelo, sabemos que o momento fletor é definido de acordo com a equação (2.16).

$$M = EI \frac{\partial^2 \omega(x, t)}{\partial x^2} \quad (2.16)$$

O esforço cortante na lâmina é definido pela equação (2.17),

$$f_c = EI \frac{\partial^3 \omega(x, t)}{\partial x^3} \quad (2.17)$$

onde  $E$  é o módulo de Young,  $I$  é o momento da seção transversal e  $\omega(x, t)$  é o deslocamento transversal da lâmina.

Tomando-se a função de deslocamento, equação (2.3), que na sua forma discreta torna-se a equação (2.18),

$$\omega_f(x(i), t(n)) = A \left( 1 - \exp^{-(XX(i) - X_{m1})} \right) \left( 1 - \exp^{(XX(i) - X_{m1}) - L} \right) \operatorname{sen} \left( \left( \frac{\pi}{a} \right) (XX(i) - X_{m1}) - \phi \Delta t_d \right), \quad (2.18)$$

onde  $(X_{m1}, Y_{m1})$  são as coordenadas da origem, ou seja, do ponto na malha que corresponde a fixação da extremidade da lâmina, e  $(XX, YY)$  são as coordenadas de um ponto da malha onde deve se localizar a fronteira móvel.

Utilizando a equação (2.18) e a equação (2.17), para  $x(i)$  e  $x(i+1)$  correspondentes aos limites do  $i$ -ésimo elemento da lâmina, resultam as expressões das forças cortantes,

$$\begin{aligned}
f_c^- = & \left[ Ae^{-x(i)}(1 - e^{x(i+1)-L})\text{sen} \left( \frac{\pi}{a}x(i) - \phi\Delta tn \right) \right] EI + \\
& \left[ -3Ae^{-x(i)}(1 - e^{x(i+1)-L})\text{cos} \left( \frac{\pi}{a}x(i) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right) \right] EI + \\
& \left[ -6Ae^{-x(i)}(e^{x(i+1)-L})\text{cos} \left( \frac{\pi}{a}x(i) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right) \right] EI + \\
& \left[ -3Ae^{-x(i)}(1 - e^{x(i+1)-L})\text{sen} \left( \frac{\pi}{a}x(i) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right)^2 \right] EI + \\
& \left[ -A(1 - e^{-x(i)})(e^{x(i+1)-L})\text{sen} \left( \frac{\pi}{a}x(i) - \phi\Delta tn \right) \right] EI + \\
& \left[ -3A(1 - e^{-x(i)})(e^{x(i+1)-L})\text{cos} \left( \frac{\pi}{a}x(i) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right) \right] EI + \\
& \left[ 3A(1 - e^{-x(i)})(e^{x(i+1)-L})\text{sen} \left( \frac{\pi}{a}x(i) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right)^2 \right] EI + \\
& \left[ -A(1 - e^{-x(i)})(1 - e^{x(i+1)-L})\text{cos} \left( \frac{\pi}{a}x(i) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right)^3 \right] EI
\end{aligned} \tag{2.19}$$

e,

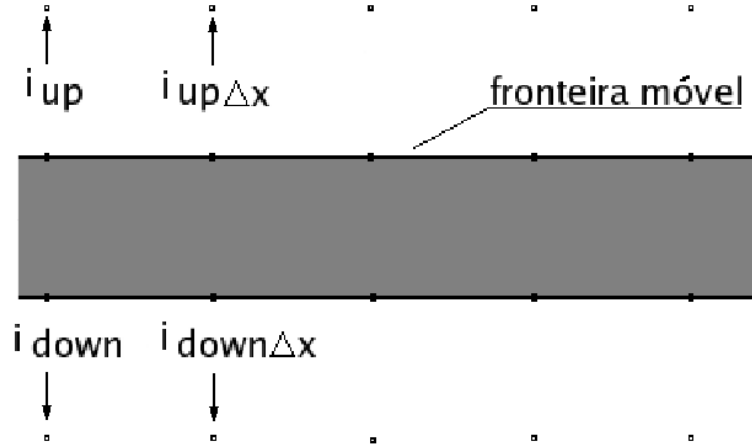
$$\begin{aligned}
f_c^+ = & \left[ Ae^{-x(i+1)}(1 - e^{x(i+1)-L})\text{sen} \left( \frac{\pi}{a}x(i+1) - \phi\Delta tn \right) \right] EI + \\
& \left[ -3Ae^{-x(i+1)}(1 - e^{x(i+1)-L})\text{cos} \left( \frac{\pi}{a}x(i+1) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right) \right] EI + \\
& \left[ -6Ae^{-x(i+1)}(e^{x(i+1)-L})\text{cos} \left( \frac{\pi}{a}x(i+1) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right) \right] EI + \\
& \left[ -3Ae^{-x(i+1)}(1 - e^{x(i+1)-L})\text{sen} \left( \frac{\pi}{a}x(i+1) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right)^2 \right] EI + \\
& \left[ -A(1 - e^{-x(i+1)})(e^{x(i+1)-L})\text{sen} \left( \frac{\pi}{a}x(i+1) - \phi\Delta tn \right) \right] EI + \\
& \left[ -3A(1 - e^{-x(i+1)})(e^{x(i+1)-L})\text{cos} \left( \frac{\pi}{a}x(i+1) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right) \right] EI + \\
& \left[ 3A(1 - e^{-x(i+1)})(e^{x(i+1)-L})\text{sen} \left( \frac{\pi}{a}x(i+1) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right)^2 \right] EI + \\
& \left[ -A(1 - e^{-x(i+1)})(1 - e^{x(i+1)-L})\text{cos} \left( \frac{\pi}{a}x(i+1) - \phi\Delta tn \right) \left( \frac{\pi}{a} \right)^3 \right] EI
\end{aligned} \tag{2.20}$$

### 2.3.2.2 Forças provenientes da pressão

As pressões médias na face superior e inferior ao  $i$ -ésimo elemento da lâmina são obtidas numericamente a partir do campo de pressão, utilizando as equações 2.21 e 2.22, ver também a Figura (2.5).

$$\bar{P}^{-}(i) = \frac{\bar{P}(i_{down}) + \bar{P}(i_{down\Delta x})}{2} \quad (2.21)$$

$$\bar{P}^{+}(i) = \frac{\bar{P}(i_{up}) + \bar{P}(i_{up\Delta x})}{2} \quad (2.22)$$



**Figura 2.5:** Nós empregados no cálculo da pressão no  $i$ -ésimo elemento de lâmina.

### 2.3.2.3 Forças de inércia

O elemento de massa  $\Delta m$  é definido pela equação (2.23), onde  $\Delta l$  é a espessura da lâmina e  $\Delta z$  é o comprimento de um elemento de lâmina.

$$\Delta m = \rho_{lamina}(\Delta l)(\Delta z) \quad (2.23)$$

A aceleração do  $q$ -ésimo elemento é a derivada segunda da função deslocamento empregando o método das diferenças centrais, ver a equação (6.3) e equação (2.3).

## 2.4 Desenvolvimento do modelo experimental

Para avaliar o modelo numérico deve ser construído um modelo experimental, uma *prova de conceito*. Os resultados experimentais podem ser utilizados para ajustar o modelo numérico.

Para garantir a semelhança física entre modelo numérico e experimental, adotou-se semelhança geométrica, as dimensões do modelo experimental são as

mesmas que as do modelo numérico<sup>3</sup>. O fluido de trabalho adotado foi a água destilada, sendo definida a viscosidade cinemática igual a  $0,993 \times 10^{-6} \text{ [m}^2/\text{s]}$  para a temperatura de  $20^\circ\text{C}$ .

Serão utilizados atuadores eletromagnéticos para deslocar a lâmina. A intensidade de corrente em cada atuador será determinada por um controlador que minimiza o erro de posição definida pela equação (2.3). As características não lineares dos atuadores eletromagnéticos devem ser levadas em consideração, através de uma tabela que relaciona força com corrente. A idéia piloto do projeto está representada na Figura (2.8).

### 2.4.1 Atuadores eletromagnéticos

O núcleo dos atuadores eletromagnéticos deve ser constituído por lâminas de ferro silício de grãos orientados sobrepostas, Núcleos... (2005, Acesso em 13 de fevereiro de 2005). Cada lâmina é recoberta com uma fina camada de epoxi que deve manter as lâminas unidas e eletricamente isoladas.

A força de atração, produzida por um solenoide é dada pela equação (2.24) Boffi L. V. (1977), na qual se observa a relação não linear entre a força  $f$  e a corrente  $i$ , assim como entre a força  $f$  e a distância  $y$ .

$$f_{mag} = \mu_0^2 \frac{n_e^2 i^2}{4} \frac{ld}{y^2} \quad (2.24)$$

onde  $\mu_0$  é a permeabilidade magnética do ar,  $n_e$  é número de espiras,  $i$  é a corrente elétrica,  $l$  é a dimensão de um lado do núcleo,  $d$  é a dimensão do lado complementar da bobina e  $y$  é a distância entre o núcleo e o entre-ferros, no caso, a lâmina.

O valor de  $\mu_0$  para o vácuo é de  $4\pi \times 10^{-7} \text{ [N/A}^2\text{]}$ . Entretanto, para a bomba a permeabilidade que interessa é a do ferro silício, do ar, do alumínio, da água e do aço inoxidável, resultando na equação (2.25).

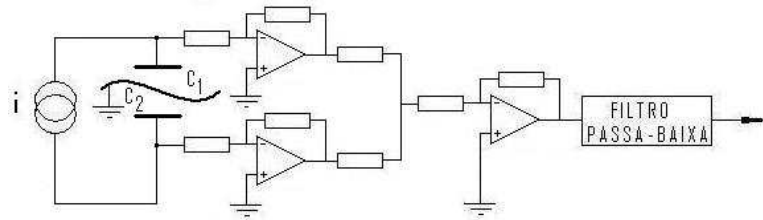
$$f_{mag} = \mu_{bomba}^2 \frac{n_e^2 i^2}{4} \frac{l^2}{y^2} \quad (2.25)$$

---

<sup>3</sup>Há relação de escala 1:1 entre modelo e protótipo

### 2.4.2 Sensor de posição

A posição da lâmina será avaliada por um sensor capacitivo diferencial. Seus eletrodos são duas placas de cobre de  $2\text{cm}^2$  de área, revestidos com uma fina camada de ouro 18 *quilates* para evitar oxidação do cobre, fixadas a paredes opostas do duto. Sobre as placas deve ser aplicada uma corrente elétrica alternada na forma de duas senoides defasadas em  $180^\circ$ , cada fase sobre um eletrodo. A diferença de potencial elétrico é medido entre os eletrodos e a própria lâmina, cujo potencial é nulo<sup>4</sup>, representada na Figura (2.6) pela placa central móvel, que oscila entre as placas fixas. O dielétrico é o fluido de trabalho que escoa entre as placas fixas e a placa móvel.



**Figura 2.6:** Sensor de posição diferencial capacitivo teórico.

Enquanto a lâmina permanecer fixa na posição central deve haver um equilíbrio entre os capacitores formados entre os eletrodos e a lâmina. Quando a lâmina se afastar da posição central passa a haver um desequilíbrio entre as capacitâncias. A variação diferencial entre as capacitâncias é dada pela equação (2.26). A capacitância total é dada pela equação (2.27).

$$\delta C = \frac{\epsilon_r \epsilon_0 S \delta y}{y^2 + \delta y^2} \quad (2.26)$$

$$C = \frac{\epsilon_r \epsilon_0 S y}{y^2 + \delta y^2} \quad (2.27)$$

onde  $C$  é a capacitância,  $S$  é a área das placas,  $y$  é a distância entre as placas,  $\epsilon_r$  é a constante dielétrica relativa,  $\epsilon_0$  é a constante dielétrica do vácuo.

De acordo com Webster (1999), a capacitância  $C$  é governada pela equação (2.28), obtida diretamente pela relação entre as equações (2.26) e (2.27), e “a saída deste transdutor é não linear com respeito a  $y$  tendo uma função de transferência hiperbólica”.

<sup>4</sup>terra

$$\frac{\delta C}{\delta y} = \frac{C}{y} \implies \frac{\delta C}{C} = \frac{\delta y}{y} \quad (2.28)$$

Esta equação permite verificar que a sensibilidade do sensor é inversamente proporcional a distância entre lâmina e placa, mas a variação de  $C$  é proporcional a variação de  $y$ . Circuitos e aplicações de sensores capacitivos podem ser vistos em Baxter (1997).

**Tabela 2.1:** Permissividade elétrica ou constante dielétrica.

Descrição	Símbolo	Valor $[\frac{F}{m^{-1}}]$
$\epsilon$ relativa do ar ou do vácuo	$\epsilon_r$	1
$\epsilon$ do vácuo	$\epsilon_0$	$8.854188 \times 10^{-12}$
$\epsilon$ da água destilada	$\epsilon_w$	80

#### 2.4.2.1 Construção do sensor capacitivo

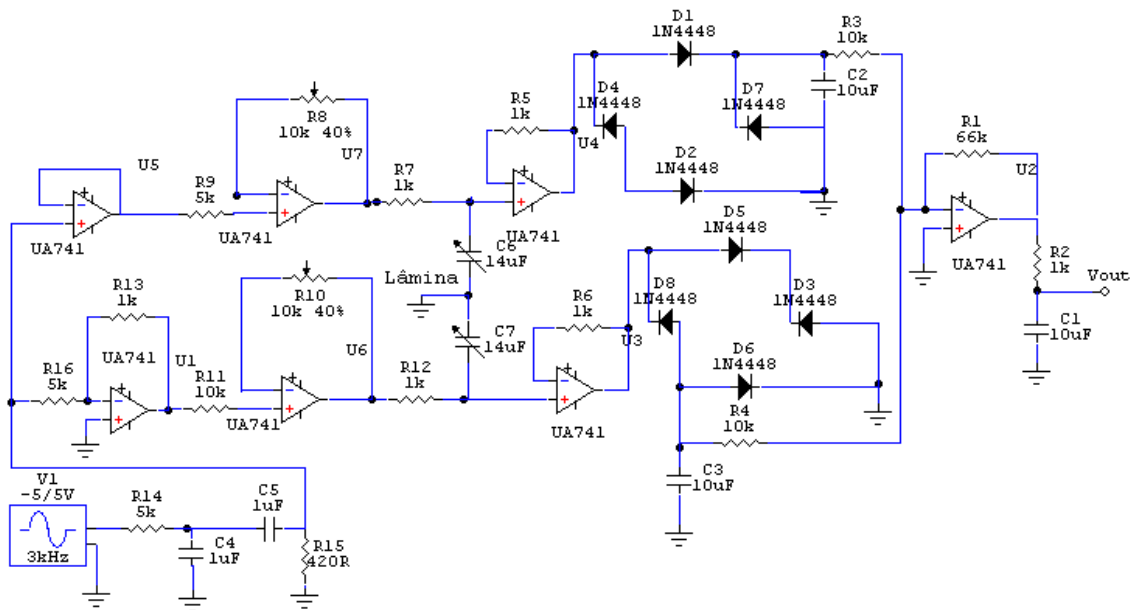
Sobre as placas de cada sensor capacitivo deve ser aplicada uma corrente alternada com amplitude de  $\pm 5V$ . Para gerar corrente alternada pode ser utilizado uma combinação de circuitos, uma fonte de tensão alternada Mims (2004), um conversor de tensão em corrente Horovitz e Hill (1989) e um amplificador de alta impedância Baxter (1997). Estes circuitos foram combinados, adaptados e ajustados, de modo a se obter o circuito de tratamento dos sinais dos sensores, apresentados na Figura (2.7).

Considerando que a área é constante e  $\epsilon$  também é constante, mas a distância  $d$  entre as placas é variável em função do tempo, a capacitância deve variar. A diferença na tensão entre os capacitores permite identificar a posição da lâmina.

#### 2.4.3 A lâmina propulsora

As lâminas flexíveis para a construção do atuador flexível serão obtidas a partir de calibradores de folga<sup>5</sup>. Seu comprimento e largura serão ajustados para as

<sup>5</sup>marca *Starret* modelo 667M-20



**Figura 2.7:** Esquema eletrônico do circuito de tratamento do sinal dos sensores diferenciais capacitivos.

dimensões do canal da bomba, mas a espessura deve ser mantida. Verifica-se, através de medição da força de atração eletromagnética, que a lâmina apresenta perda de material ferromagnético e progressiva diminuição da força de resposta ao campo do atuador. Torna-se necessário revestir a lâmina com um material que impessa a perda de material ferromagnético mas que mantenha a condutividade elétrica ao longo da lâmina, uma vez que ela atua como terra dos sensores capacitivos diferenciais. Tentou-se empregar uma tinta condutora a base de grafite, aplicada em cinescópios, entretanto, ela se mostrou excessivamente espessa e quebradiça. O problema foi contornado revestindo a lâmina com uma fina camada de ouro. Posteriormente a lâmina foi revestida com Teflon, para minimizar o fluxo de corrente entre eletrodos e lâmina.

#### 2.4.4 Estratégia de controle do movimento da lâmina

Para verificar a controlabilidade e a observabilidade do estado do sistema, deve ser desenvolvido um modelo estrutural da lâmina para projetar um controlador do deslocamento da lâmina. Este modelo estrutural considera apenas a inércia e a rigidez da lâmina. Nesta seção apresenta-se um modelo em espaço de estados da lâmina, obtido a partir de um modelo estrutural MEF, os testes de observabilidade e controlabilidade do sistema e a definição do controlador. Caso o sistema não se apresente controlável ou observável, novas configurações de sensores e atuadores devem ser investigadas.

### 2.4.4.1 Modelo dinâmico da lâmina

Empregou-se um modelo de elementos finitos, no qual se dividiu a lâmina em seis elementos. O número de elementos foi escolhido de modo que cada nó entre elementos estivesse na posição sobre a qual será aplicada a resultante da força eletromagnética de cada bobina. Ver Figuras (2.8) e (2.9).

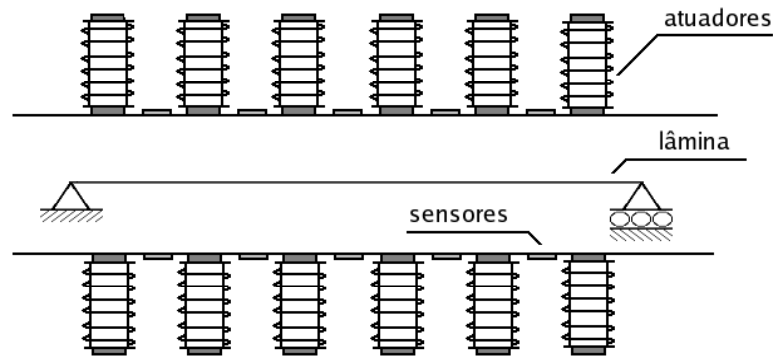


Figura 2.8: Representação da lâmina, sensores e atuadores.

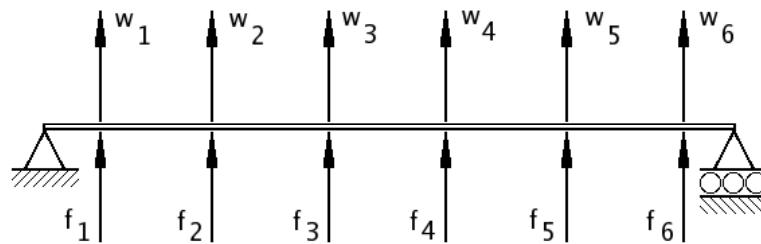


Figura 2.9: Modelo dinâmico de elementos finitos da lâmina.

O modelo de controle se limita a controlar os deslocamentos transversais, na direção  $y$ . Os deslocamentos longitudinais, na direção  $x$ , de um ponto qualquer da lâmina não são calculados. A partir do modelo de elementos finitos estrutural FELT, se obtém a matriz consistente de massa e a matriz de rigidez global. Estas matrizes permitem a obtenção de um modelo de controle no espaço de estados. Do MEF sabe-se que o modelo de estrutura segue a equação (2.29).

$$[\mathbf{M}] \ddot{x} + [\mathbf{K}] x = \{F_n\} \quad (2.29)$$

onde  $\mathbf{M}$  é a matriz de massa e  $\mathbf{K}$  a matriz de rigidez, sendo  $\dot{x}$  o vetor de velocidades nodais,  $x$  o vetor dos deslocamentos nodais e  $F_n$  o vetor das forças nodais.

Os elementos numéricos das matrizes  $\mathbf{M}$  e  $\mathbf{K}$  podem ser vistos no *APÊNDICE*



*B* - Algoritmos em linguagem *C*, e representam a discretização do modelo para  $\Delta t^6$  que corresponde a frequência de excitação do terceiro modo natural de vibração da lâmina<sup>7</sup>.

Definindo-se o vetor de estado  $\chi$  e sua derivada temporal  $\dot{\chi}$ , conforme a equação (2.30) e equação (2.31).

$$\chi = \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} \quad (2.30)$$

$$\dot{\chi} = \begin{Bmatrix} \dot{x} \\ \ddot{x} \end{Bmatrix} \quad (2.31)$$

Resulta a equação (2.32),

$$\begin{Bmatrix} \dot{x} \\ \ddot{x} \end{Bmatrix} = \begin{bmatrix} 0 & I \\ -\mathbf{M}^{-1}\mathbf{K} & 0 \end{bmatrix} \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} + \begin{bmatrix} 0 \\ -\mathbf{M}^{-1} \end{bmatrix} \{F\} \quad (2.32)$$

ou, na linguagem normalmente utilizada em teoria de controle linear, o sistema é descrito por matrizes  $\mathbf{A}$ ,  $\mathbf{B}$ , conforme a equação (2.33). A matriz de observações  $\mathbf{C}$  está definida na equação (2.34) e na equação (2.38).

$$\dot{x} = \mathbf{A}x + \mathbf{B}u \quad (2.33)$$

$$y = \mathbf{C}x \quad (2.34)$$

As matrizes globais foram obtidas por superposição das matrizes locais de cada elemento finito do modelo de lâmina. Antes que essas fossem empregadas, na determinação das matrizes  $\mathbf{A}$  e  $\mathbf{B}$ , as condições de contorno foram impostas, ver equações (2.35).

$$\begin{aligned} y(0) &= 0; & y(L) &= 0; \\ \dot{y}(0) &= 0; & \dot{y}(L) &= 0; \end{aligned} \quad (2.35)$$

As condições de contorno reduziram a ordem das matrizes  $\mathbf{M}$  e  $\mathbf{K}$  e permitiram a determinação das matrizes  $\mathbf{A}$ ,  $\mathbf{B}$  e  $\mathbf{C}$  através das equações (2.36), (2.37) e (2.38).

---

<sup>6</sup> $\Delta = 0,005s$

<sup>7</sup>o que garante o controle dos três primeiros modos de vibrar da lâmina

$$\mathbf{A}_{(24 \times 24)} = \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{M}_r^{-1} \mathbf{K}_r & 0 \end{bmatrix} \quad (2.36)$$

$$\mathbf{B}_{(24 \times 12)} = \begin{bmatrix} 0 \\ -\mathbf{M}_r^{-1} \end{bmatrix} \quad (2.37)$$

$$\mathbf{C}_{(10 \times 24)} = \begin{bmatrix} \mathbf{I} & 0 \\ 0 & 0 \end{bmatrix} \quad (2.38)$$

#### 2.4.4.2 Controlabilidade e observabilidade

A partir das matrizes  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  e do intervalo de discretização  $\Delta t$  foram calculadas as matrizes do sistema no tempo discreto,  $\mathbf{a}_d$ ,  $\mathbf{b}_d$  e  $\mathbf{c}_d$ . A condição de controlabilidade discreta, segundo Ogata (1994) requer que a matriz de controlabilidade tenha posto completo, ou seja:

$$[\mathbf{b}_d \ \mathbf{a}_d \mathbf{b}_d \ \dots \ \mathbf{a}_d^{n-1} \mathbf{b}_d] \quad (2.39)$$

Deve ter posto completo, condição que foi satisfeita.

A condição de observabilidade discreta não foi inicialmente alcançada. A partir do modelo MEF proposto para a lâmina, é possível que tenha ocorrido um posicionamento de algum sensor sobre algum nó de um modo natural de vibrar da lâmina. Neste caso, os sensores não detectariam o deslocamento da lâmina neste modo de vibrar. Para contornar essa situação, se fez com que o primeiro e o último elementos do modelo de elementos finitos tivessem comprimentos maiores que os elementos centrais. Após essa alteração, o sistema passou a apresentar observabilidade discreta.

#### 2.4.4.3 Definição do controlador

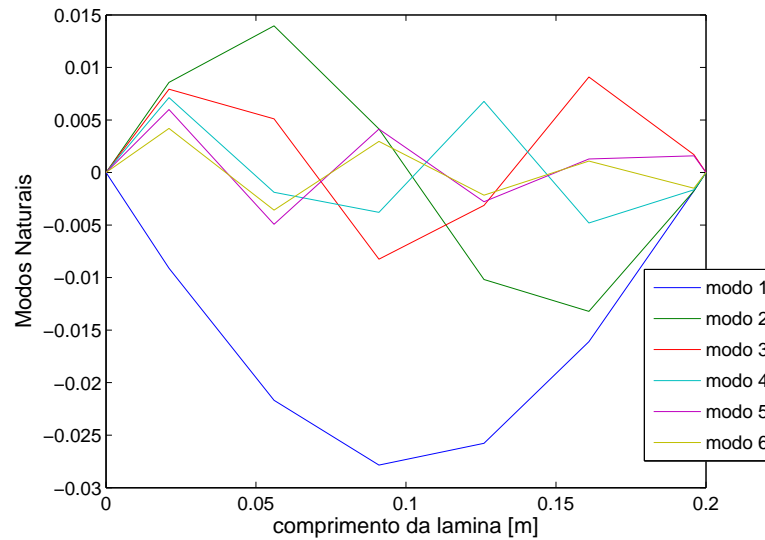
Será construído um controlador linear e invariante no tempo por realimentação de estados. A lei de controle segue a equação (2.40).

$$u = K_g(x_{ref} - x_{med}) \quad (2.40)$$

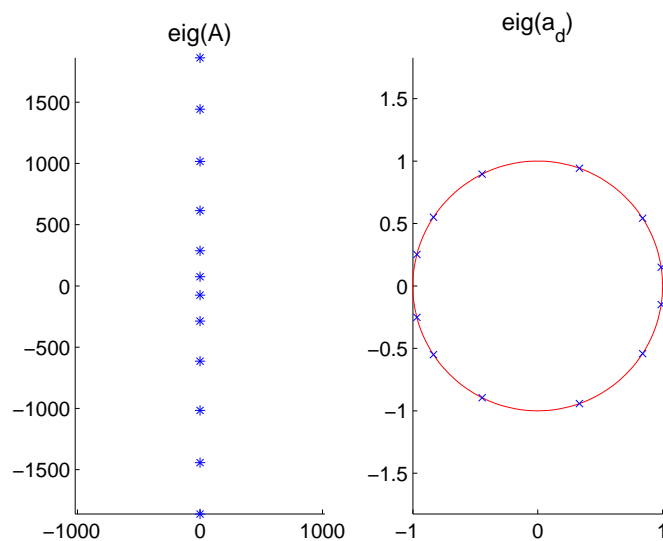
onde A matriz de ganho  $K_g$  é diagonal e os elementos de sua diagonal são iguais.  $K$

é a matriz de ganho,  $u$  é o sinal de controle,  $x_{ref}$  é o vetor de deslocamentos de referência e  $x_{med}$  é o vetor de deslocamentos medidos. A matriz de ganho  $K_g$  é diagonal e os elementos de sua diagonal são iguais.

Na Figura (2.10) estão representados os seis primeiros modos naturais de vibração da lâmina. Deve-se evitar posicionar sensores ou atuadores nos nós destes modos. Na Figura (2.11) estão representados os autovalores da matriz do sistema  $A$ , os autovalores da matriz discreta do sistema  $a_d$ .



**Figura 2.10:** Seis primeiros modos naturais de vibração para o modelo MEF da lâmina.



**Figura 2.11:** Gráfico dos pólos de malha aberta.

## 3 Resultados

### 3.1 Resultados do modelo numérico

Há uma solução de compromisso entre o tempo de computação da solução e a resolução da resposta. A malha de elementos finitos empregada, utilizou 6369 nós, produzindo um arquivos de 1,2 Mb. Este refinamento aparentemente excessivo da malha é necessário para garantir uma boa definição e a integridade da fronteira móvel. Se a malha tiver resolução maior, surgem descontinuidades na fronteira móvel devido a  *fina* espessura da lâmina<sup>1</sup>, que corresponde a 2% da largura do canal.

As simulações foram realizadas para um tempo total de 1s, com intervalo de tempo de 0,01s. Vinte conjuntos de variáveis de projeto foram simuladas. Foram avaliados quatro valores da amplitude de oscilação da lâmina<sup>2</sup>. Para cada amplitude foram feitas cinco simulações com diferentes frequências<sup>3</sup>, todas com comprimento de onda de 0,12m. Ver Tabela (3.1). A velocidade na seção de entrada da bomba foi realimentada numericamente pelo valor médio da velocidade na saída da bomba. A variação de forma sofrida pela lâmina pode ser vista nas Figuras (3.1) e (3.2).

**Tabela 3.1:** Parâmetros de projeto

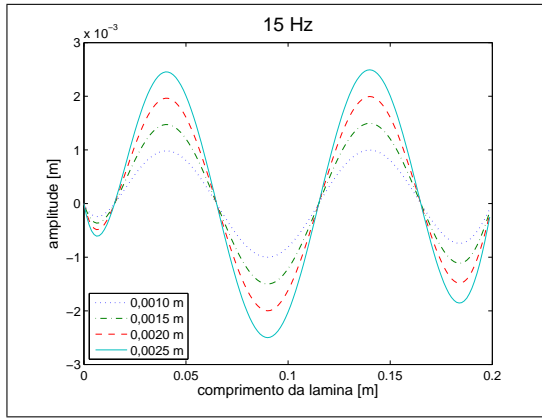
Frequências [Hz]	Amplitudes [m]	Vels. Ang. [rad/s].
1	0,0010, 0,0015, 0,0020, 0,0025	$2\pi, 10\pi, 20\pi, 30\pi, 40\pi$
5	0,0010, 0,0015, 0,0020, 0,0025	$2\pi, 10\pi, 20\pi, 30\pi, 40\pi$
10	0,0010, 0,0015, 0,0020, 0,0025	$2\pi, 10\pi, 20\pi, 30\pi, 40\pi$
15	0,0010, 0,0015, 0,0020, 0,0025	$2\pi, 10\pi, 20\pi, 30\pi, 40\pi$
20	0,0010, 0,0015, 0,0020, 0,0025	$2\pi, 10\pi, 20\pi, 30\pi, 40\pi$

O número de Reynolds é função da amplitude de deformação transversal, ver equação (3.1), com comportamento apresentado pela Figura (3.3). A Figura (3.4)

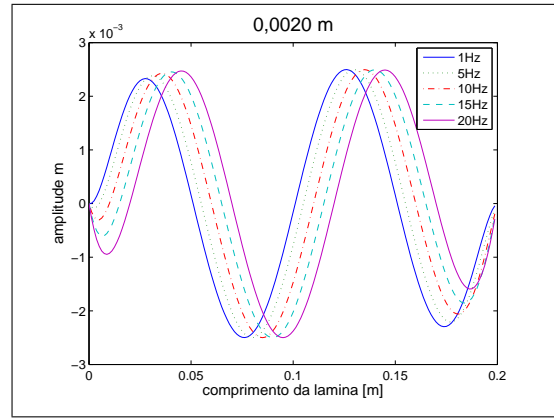
<sup>1</sup>0,0002 m

<sup>2</sup>0,0010, 0,0015, 0,0020 e 0,0025m

<sup>3</sup>1, 5, 10, 15 e 20 Hz



**Figura 3.1:** Amplitudes da senoide para uma mesma frequência e uma mesma fase.

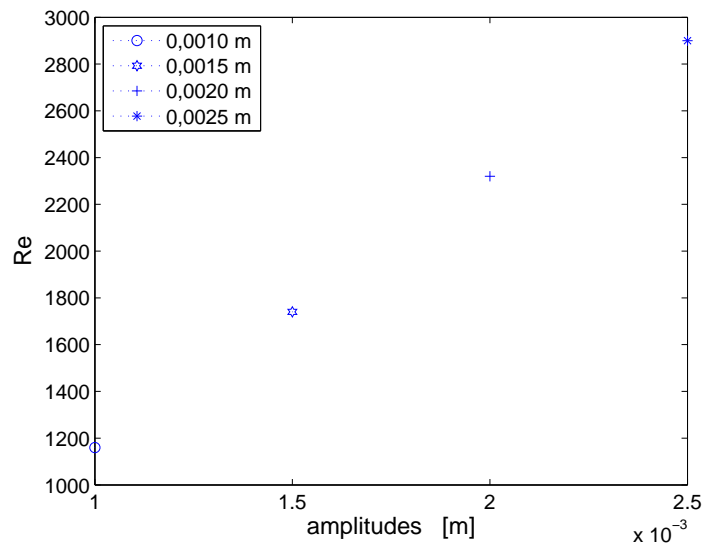


**Figura 3.2:** Deslocamento longitudinal da deformação transversal no tempo.

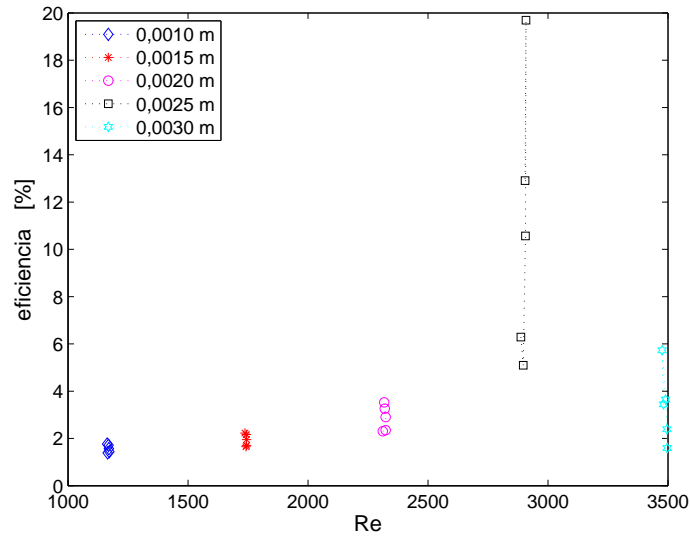
apresenta a eficiência energética em função do número de Reynolds.

$$Re = \frac{A_{mp} v}{\nu} \quad (3.1)$$

onde  $A_{mp}$  é a amplitude do deslocamento,  $Re$  é o número de Reynolds,  $v$  é a velocidade média do escoamento do fluido de trabalho e  $\nu$  a viscosidade cinemática.



**Figura 3.3:** O número de Reynolds em função da amplitude de deformação transversal da lâmina.



**Figura 3.4:** A eficiência energética em função da amplitude de deslocamento, do número de Reynolds e da frequência.

A combinação de parâmetros totalizou 20 simulações, cada simulação utiliza cerca de 4 horas em um computador pessoal PC pentium IV, 3.2 GHz, 2 Gb de RAM DDR 400 MHz e 160 Gb de memória em dois discos rígidos com comunicação serial SATA. Foram consumidos pouco mais de 80 horas para se gerar o conjunto de dados da Figura (3.19), que contém o gráfico de rendimento médio. O Featflow apresenta os resultados em dois formatos, *.gmv* e *.avs*. O programa de visualização *GMV* emprega o formato *.gmv*, enquanto que o algoritmo de cálculo do índice de desempenho utiliza resultados de saída no formato *.avs*, que é mais compacto. Resultam 200 arquivos para cada simulação.

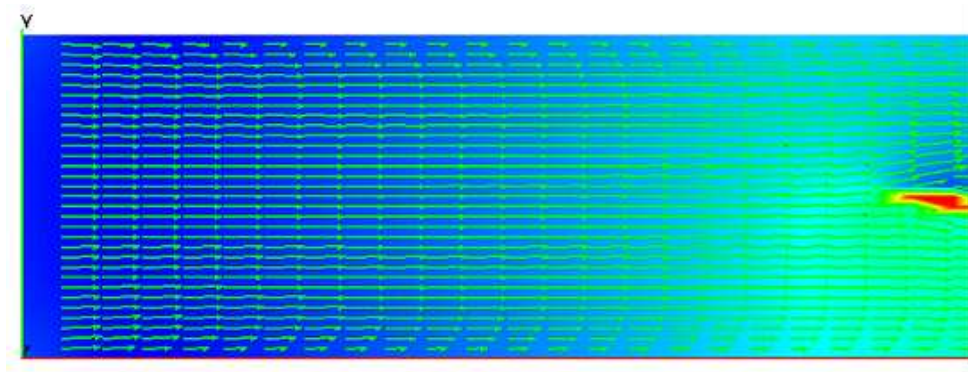
### 3.1.1 Visualização dos resultados

A visualização da solução da equação (2.4) de *Navier-Stokes* é apresentada na forma de um campo de pressão, visto na Figura (3.7) e de um campo de velocidades, mostrado na Figura (3.8). Observa-se que o escoamento ocorre da esquerda para a direita. As unidades estão em Pa e [mca]<sup>4</sup>. Nota-se no gráfico da velocidade a região de velocidade nula correspondente à fronteira móvel que representa o movimento da lâmina.

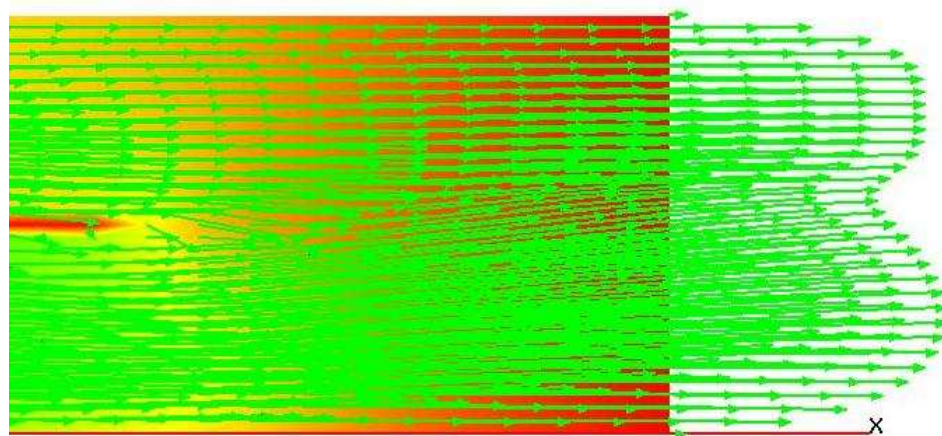
O programa de visualização gráfica permite acompanhar o desenvolvimento do perfil de velocidades ao longo do canal do modelo numérico, como na Figura (3.5) e na Figura (3.6) onde se observa o perfil de velocidade na seção de saída do

<sup>4</sup>metros de coluna d'agua

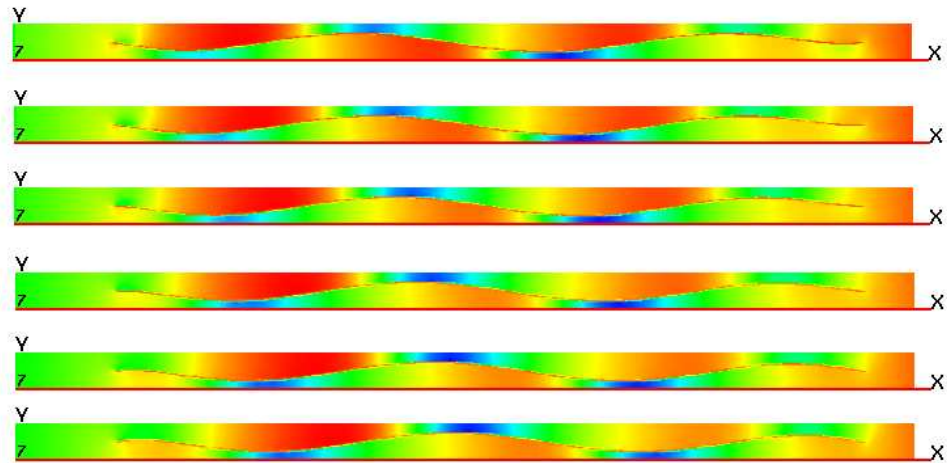
modelo numérico. Também é possível visualizar a evolução temporal dos campos de pressão, como na Figura (3.7).



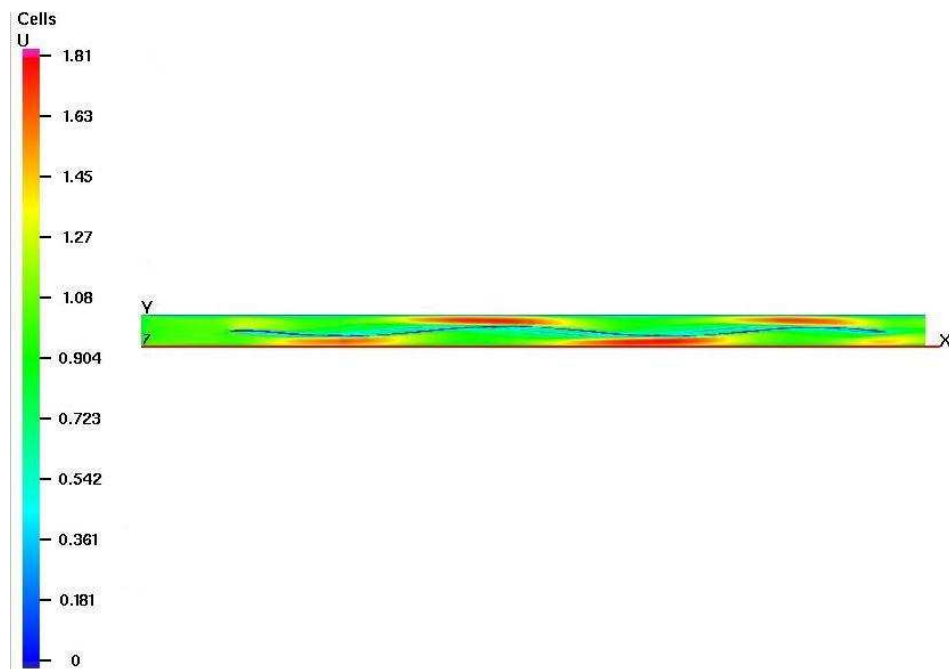
**Figura 3.5:** Perfil do escoamento da velocidade na seção de entrada ainda não desenvolvido.



**Figura 3.6:** Perfil do escoamento da velocidade na seção de saída ainda não desenvolvido.



**Figura 3.7:** Campo de pressões para frequência de 5 Hz, de 0,05 a 0,010s, de cima para baixo.

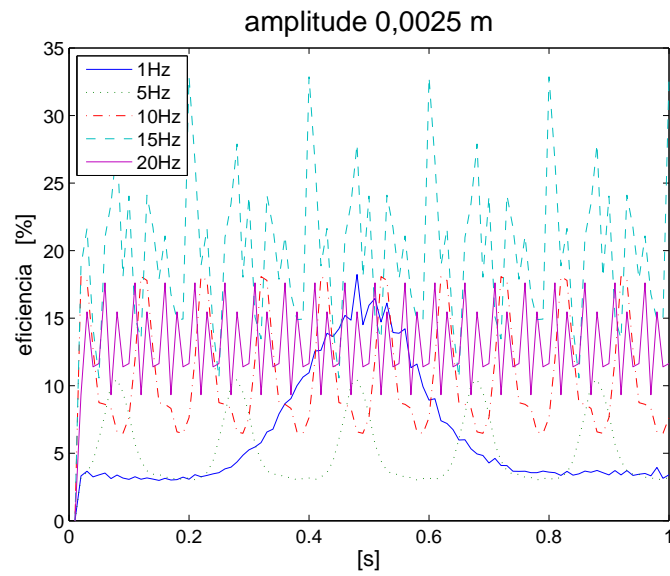


**Figura 3.8:** Campo de velocidade em  $m/s$  na direção  $x$  para frequência de 15Hz, fluxo da esquerda para a direita.

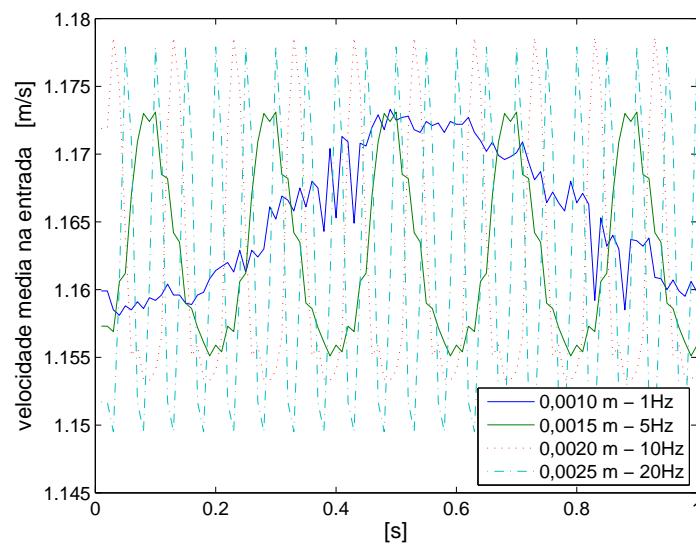
Tomando um grupo de simulações para a amplitude de 0.25 cm, podemos ver na Figura (3.9) a eficiência mecânica ao longo do tempo de simulação. Verifica-se a característica pulsátil da bomba.

A Figura (3.10) apresenta a velocidade do fluido na seção de entrada em função do tempo, para diversos valores de amplitude e frequência. Esta variável foi definida por um algoritmo de busca. Verifica-se o comportamento pulsátil da velocidade média.





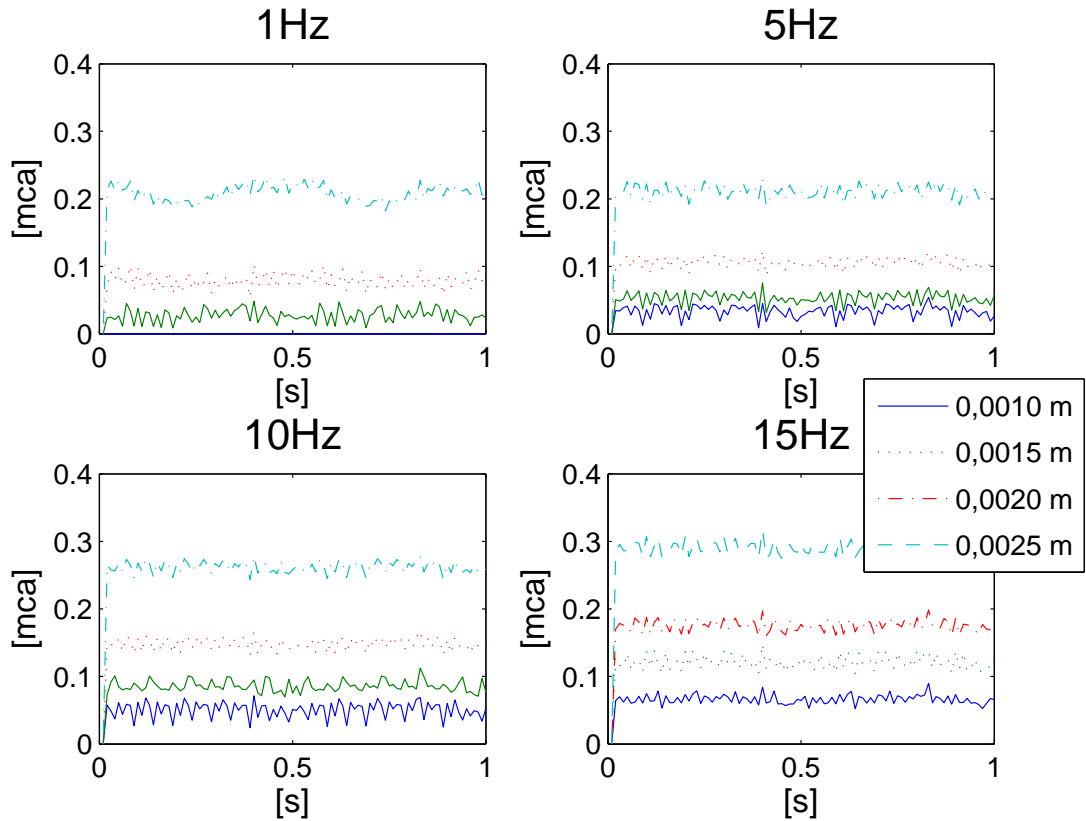
**Figura 3.9:** Comportamento pulsátil da eficiência mecânica, amplitude 0.0025m, frequências de 1 a 20Hz.



**Figura 3.10:** Velocidade média na seção de entrada versus tempo para diversos valores de frequência e amplitude.

### 3.1.2 Resposta em função da frequência

As Figuras (3.11) e (3.12), apresentam o ganho de pressão média entre as seções de saída e entrada, variando a amplitude e mantendo frequência fixa. Os valores médios correspondentes a estas figuras são apresentados na Figura (3.18).



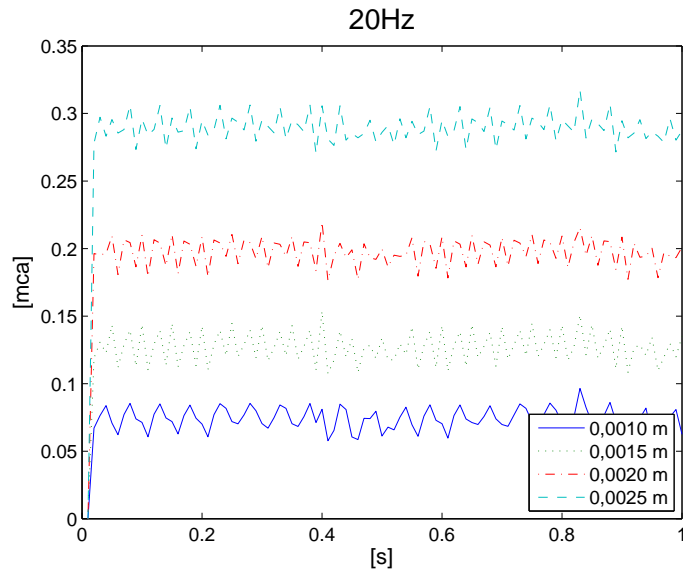
**Figura 3.11:** Ganho de pressão média versus tempo para valores de frequência de 1 a 15Hz.

As Figuras (3.13) e (3.14), apresentam a vazão, variando a amplitude e mantendo a frequência fixa.

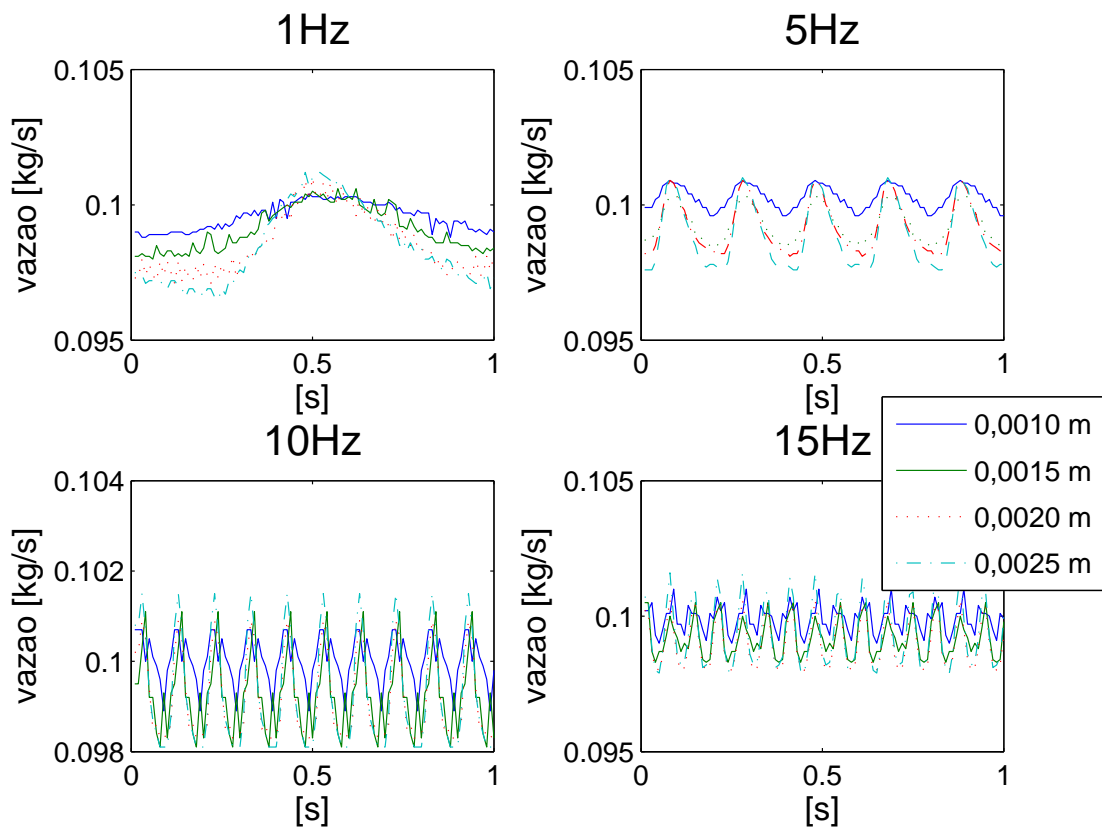
As Figuras (3.15) e (3.16), apresentam a eficiência mecânica, variando a amplitude e mantendo a frequência fixa. Os valores médios no tempo da eficiência mecânica são mostrados na Figura (3.17). Observa-se que é possível atingir eficiência mecânica média da ordem de 20%.

O ganho médio de pressão versus frequência são mostrados na Figura (3.18).

A Figura (3.19), apresenta a eficiência média para uma simulação em que a extremidade da lâmina, próxima à saída do canal não foi vinculada, permanecendo livre para se deslocar na direção  $y$ . Ao comparar a Figura (3.17) com a Figura (3.19) verifica-se que a configuração da bomba com a extremidade fixa apresenta eficiência energética máxima, cerca de 20%, que é superior à eficiência

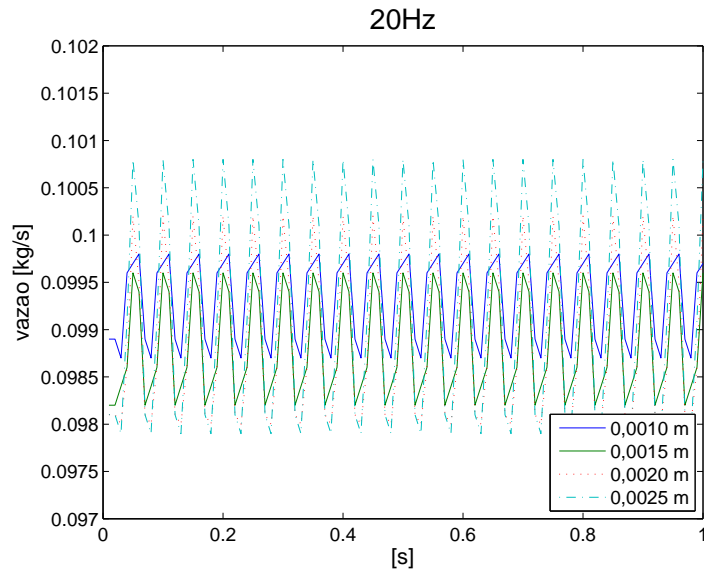


**Figura 3.12:** Ganho de pressão média versus tempo na frequência de 20Hz para valores de amplitude de deslocamento transversal de 0.0010 a 0.0025 m.

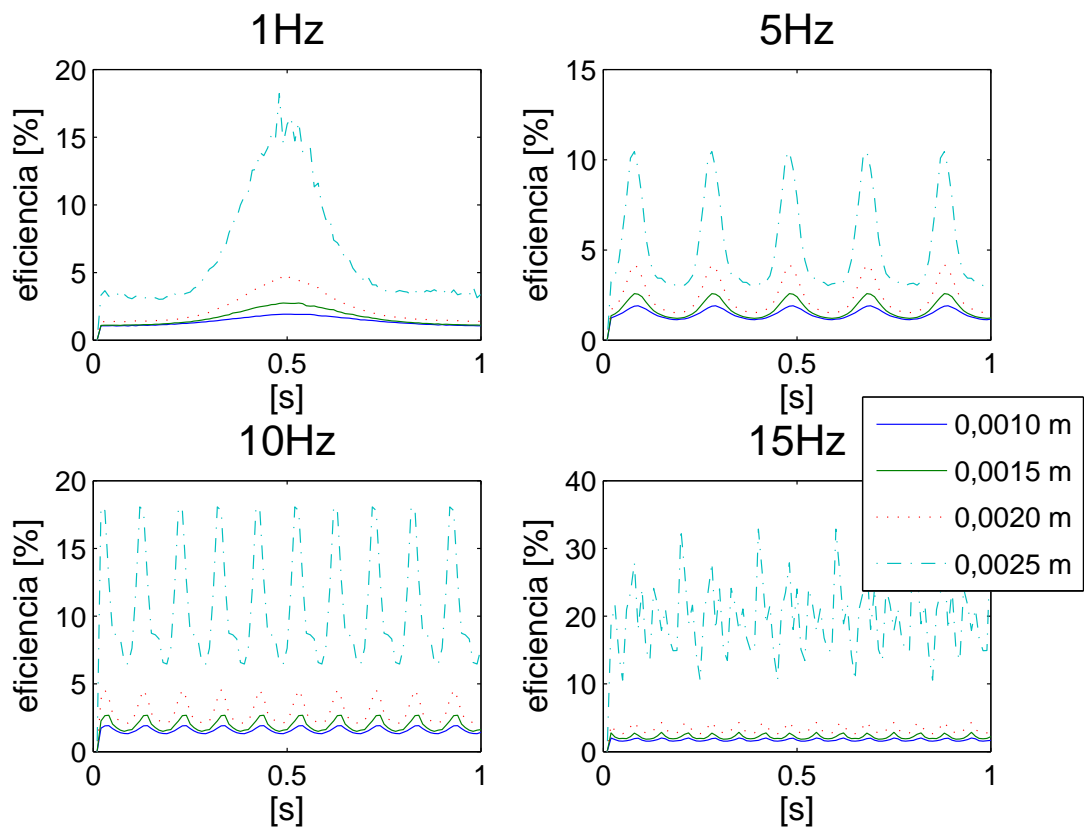


**Figura 3.13:** Vazão média versus tempo para frequências de 1 a 15Hz.

energética máxima quando a lâmina tem a extremidade próxima à saída do canal livre, que alcança eficiência mecânica máxima de aproximadamente 15%. Este resultado pode ser explicado considerando-se que a extremidade livre pode assumir posições que dificultem o fluxo e que gerem vórtices indesejados. Esta configuração foi obtida apenas modificando a função de modulação para o deslo-

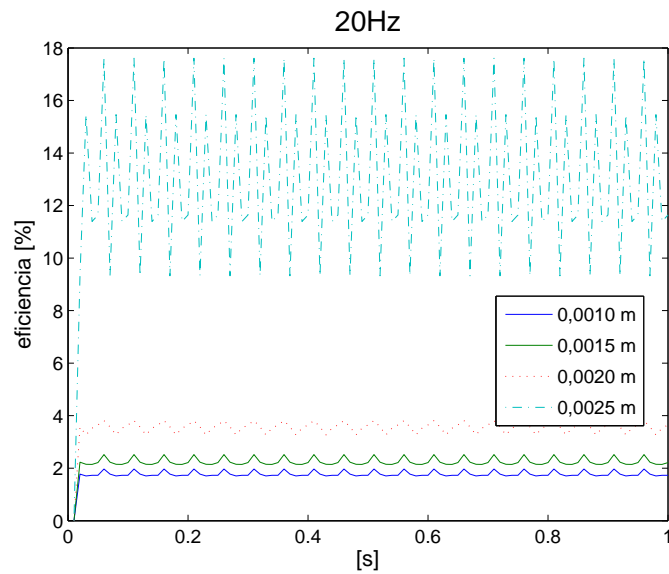


**Figura 3.14:** Vazão média versus tempo para diversas amplitudes mantendo a frequência em 20Hz.

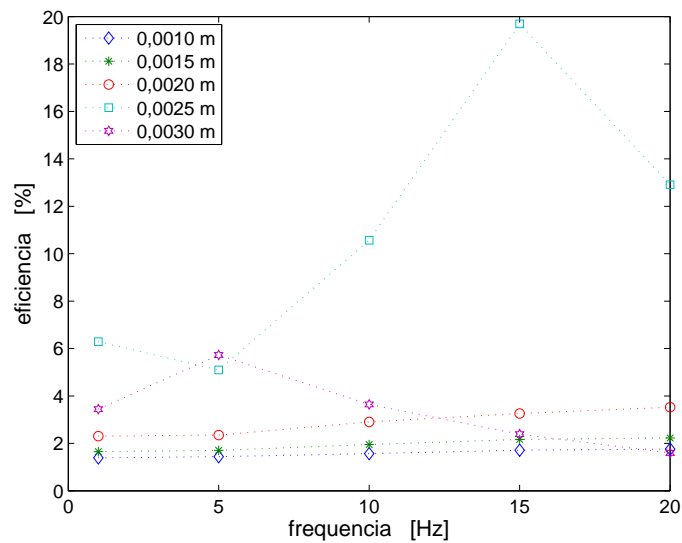


**Figura 3.15:** Eficiência mecânica média versus tempo para frequências de 1 a 15Hz.

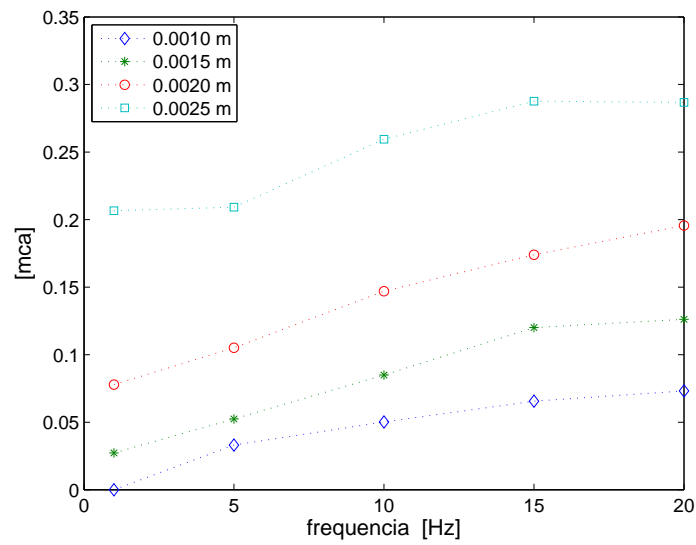
camento da lâmina, ver equação (2.3).



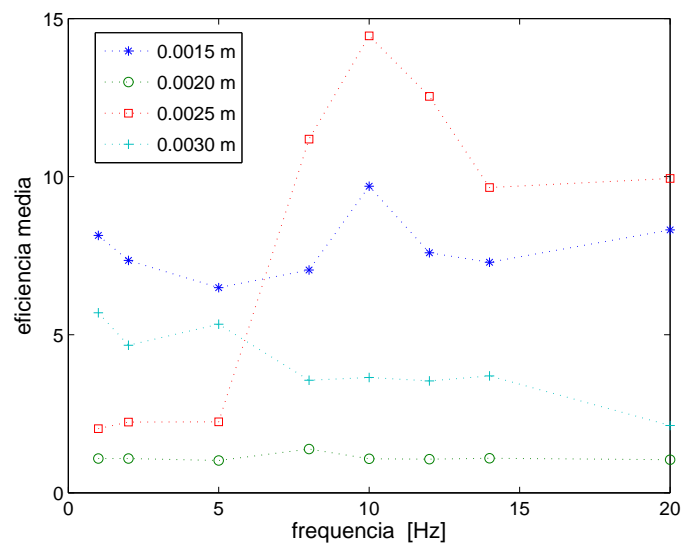
**Figura 3.16:** Eficiência mecânica média versus tempo para diversas amplitudes mantendo a frequência em 20Hz.



**Figura 3.17:** Eficiência média em função da frequência quando a extremidade da lâmina próxima à saída do canal pode apenas transladar longitudinalmente.



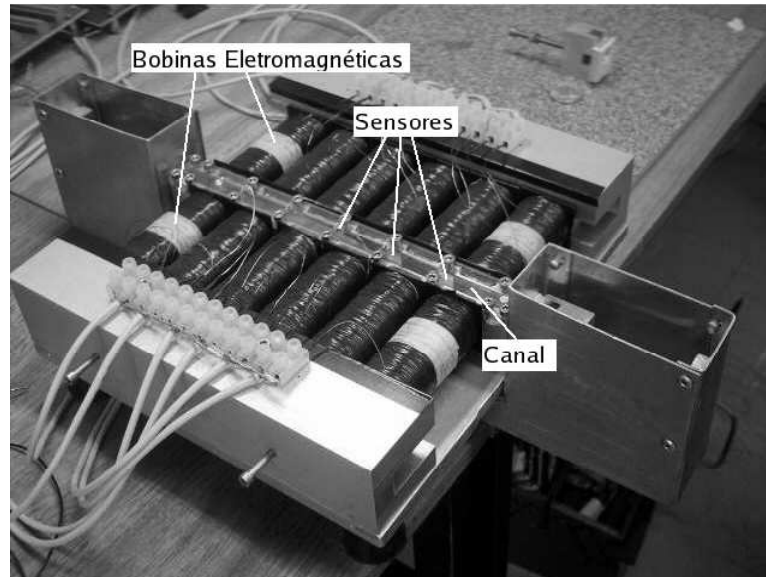
**Figura 3.18:** Ganho de pressão média versus frequência para diversas amplitudes de deslocamento transversal.



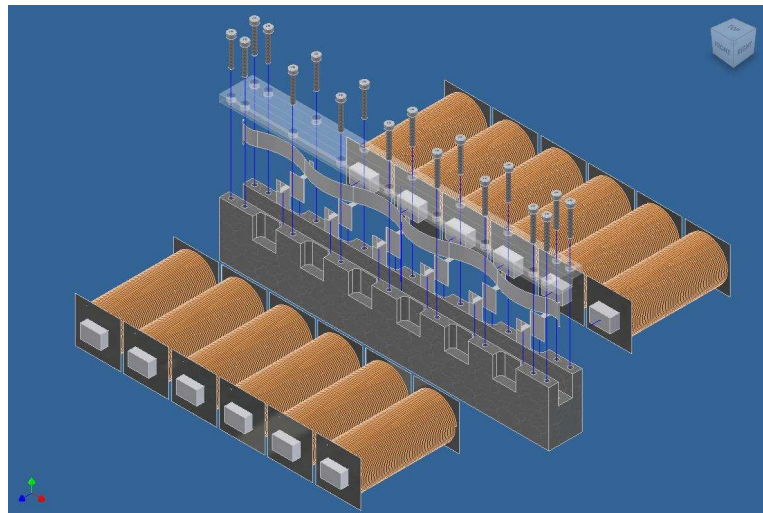
**Figura 3.19:** Eficiência média em função da frequência quando a extremidade da lâmina próxima à saída do canal está livre.

## 3.2 Resultados experimentais do protótipo

O modelo experimental foi construído, conforme a Figura 3.20. A disposição dos componentes da bomba pode ser visto na Figura 3.21. Nesta seção serão descritos os resultados de calibração, tanto dos atuadores quanto dos sensores, e testes iniciais do modelo experimental.



**Figura 3.20:** Bomba experimental com 12 atuadores eletromagnéticos ao longo do canal.

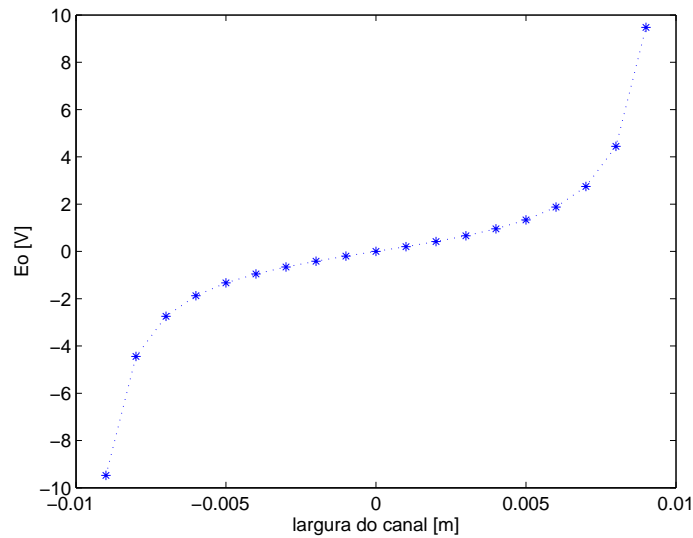


**Figura 3.21:** Vista explodida dos componentes da Bomba experimental.

### 3.2.1 Medição da posição da lâmina

Baxter (1997) apresenta, ver Figura (3.22), o sinal de potencial elétrico teórico de um sensor diferencial capacitivo. O circuito apresentado na Figura (2.7) foi

implementado para a leitura do sinal gerado nos cinco sensores de posição diferenciais capacitivos dispostos ao longo do canal. A Figura (3.23) apresenta o sinal de potencial elétrico medido pelo sensor em função da posição da lâmina. Observa-se que a faixa de tensão elétrica e a faixa de deslocamentos estão adequadas para o modelo experimental.



**Figura 3.22:** Potencial elétrico teórico de um sensor capacitivo de posição.

Com o protótipo já implementado, a medida do sinal de tensão foi refeita para cada sensor variando-se a posição da lâmina<sup>5</sup>. Cada sinal foi obtido empregando um circuito de tratamento do sinal do sensor, ver Figura (2.7). Para cada sinal obtido, obteve-se o polinômio correspondente para realimentar a malha de controle, ver Figuras (3.23) e (3.24). Deve-se verificar que o sinal lido gerou a relação *tensão X distância*, mas para efeito de controle será empregado a relação inversa *distância X tensão*, uma vez que o sinal de saída dos circuitos é a tensão e a variável de controle é a posição da lâmina.

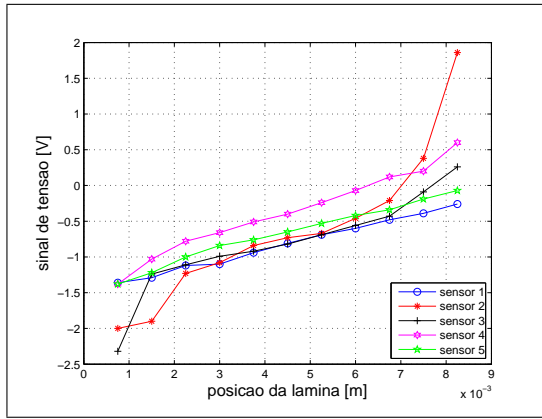
O sensor de posição mede a capacitância entre lâmina e placas metálicas fixas no canal. O sensor gera um sinal de tensão em função do deslocamento da lâmina. A lâmina é mantida no potencial nulo, terra, e mede-se o potencial entre as placas fixas no canal. Desta forma a lâmina pode ser de material condutor de corrente elétrica, por exemplo aço, e apenas uma conexão ao terra é necessária.

A realização de testes de medição do sinal de tensão referente a posição da lâmina, foram feitos com o canal aberto, ou seja sem a tampa, com a lâmina sendo deslocada manualmente com o auxílio de ferramentas.

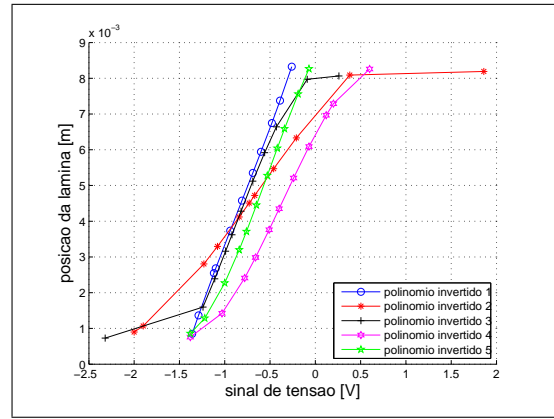
Entretanto os testes de bombeamento necessitavam que o canal estivesse

<sup>5</sup>tomando-se 11 pontos para a largura do canal, tomando-se 12 espaçadores de 0,000725 m



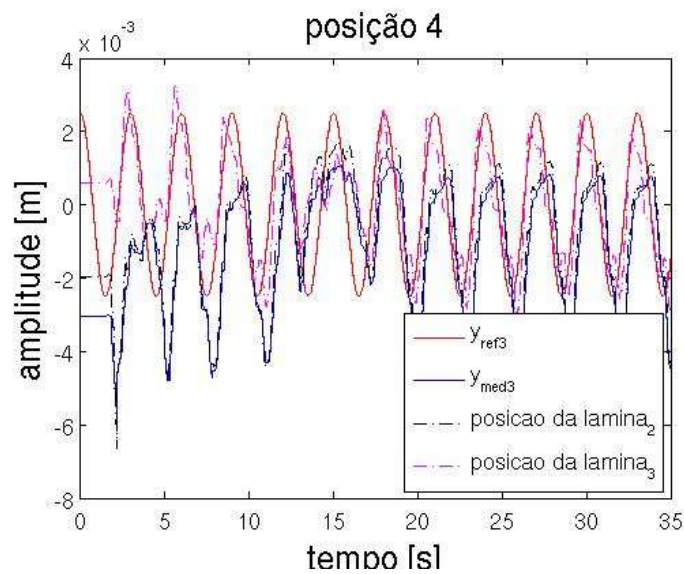


**Figura 3.23:** Sinal de tensão obtido para cada sensor de posição.



**Figura 3.24:** Sinais de tensão dos sensores invertidos e interpolados.

fechado. A colocação da tampa de acrílico no canal altera a permeabilidade magnética em torno da lâmina. Para garantir uma leitura correta da posição da lâmina, foi incorporado no algoritmo de controle da bomba um mecanismo de auto-calibração, um para cada sensor diferencial capacitivo, de modo a garantir que os valores empregados para a função de interpolação estivessem corretos. Ver Figura 3.25.



**Figura 3.25:** Comportamento do sinal de saída de um sensor diferencial capacitivo.

### 3.2.2 Medição da força de atração eletromagnética

Foram construídos doze eletromagnetos, cada um com um núcleo de ferro silício de grão orientado, com dimensões de (15 x 15 x 80) mm, constituído de lâminas de 0.35mm de espessura sobrepostas e isoladas com uma fina camada de epoxi.

O aspecto físico é apresentado na Figura (3.26).

Nas bobinas empregou-se fio de cobre com seção circular *AWG* 26 que pode conduzir em regime permanente uma corrente máxima de  $400\text{mA}$ . Nos eletromagnetos utilizaram-se 2100 espiras.



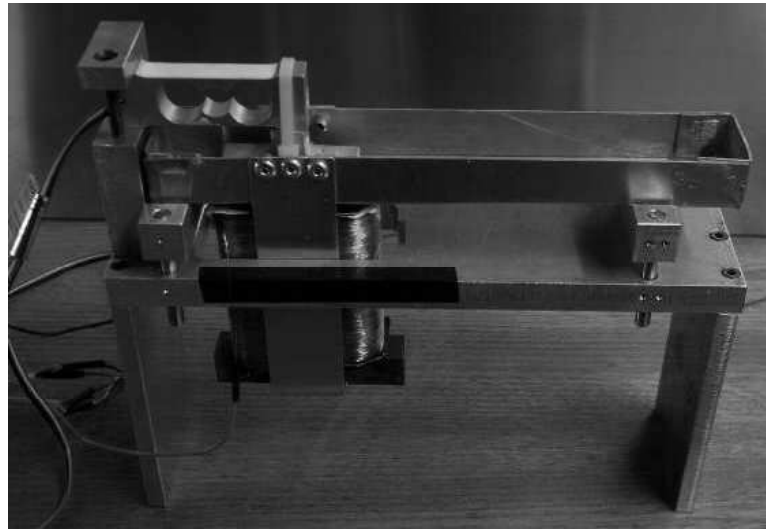
**Figura 3.26:** Aspecto final do eletromagneto.

A equação (2.24) apresenta uma relação não linear entre a força  $f$  e a variável de controle  $i$ , bem como entre a força  $f$  e a distância do entre-ferros  $y$ . Procurou-se estabelecer empiricamente a relação entre a força  $f$  e a corrente  $i$  em função da posição  $y$  da lâmina. Para isto construiu-se um dispositivo que medisse a força magnética de um par de bobinas agindo sobre a lâmina, ver Figura (3.27). Os componentes do dispositivo podem ser vistos na Figura (3.28).

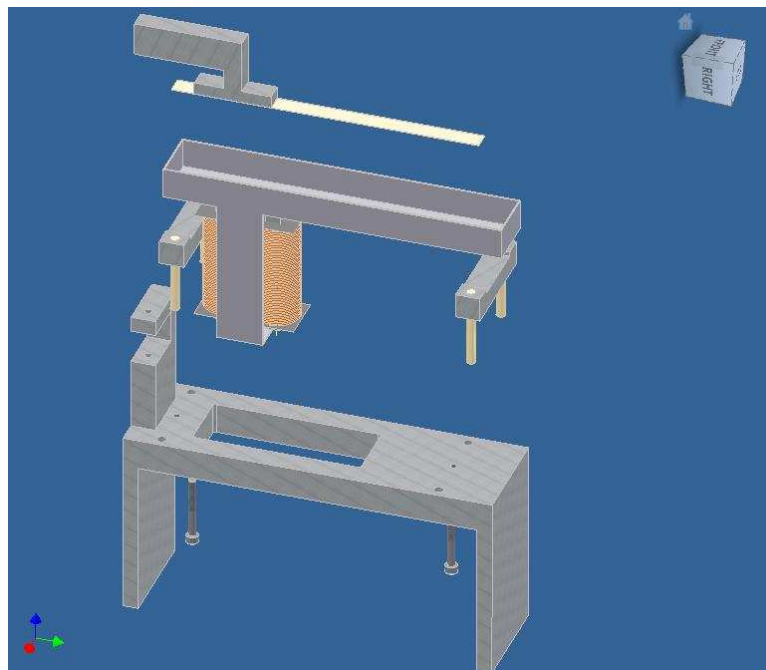
O dispositivo foi construído de modo a repetir a condição de operação da lâmina na bomba. Desta forma, possui escala de relação 1:1 com o modelo experimental. Durante todo o experimento, a lâmina permaneceu imersa em água, dentro de uma cuba de alumínio. A espessura da parede da cuba é a mesma da parede do canal da bomba.

O fluxo magnético deve emergir do núcleo de ferro silício, passar por uma fina camada de ar, atravessar a parede da cuba, atravessar o fluido de trabalho, atravessar a lâmina de aço inóx ferromagnético e, em seguida, fazer o caminho inverso, retornando ao núcleo no pólo magnético oposto.

O sensor de força empregado é uma célula de carga que utiliza uma ponte de Wheatstone completa de extensômetros, alimentada com 10V. Utilizou-se o modelo CGI para 1 kg da Células... (2006, Acesso em 02 de janeiro de 2006), com sinal de saída de  $2\text{ mV/V}$ . Esta foi calibrada, utilizando massas conhecidas para tensioná-la. Para cada massa se tomou a tensão de saída correspondente cujos resultados podem ser vistos na Figura (3.29). Estas massas foram medidas



**Figura 3.27:** Dispositivo de medição da força eletromagnética.



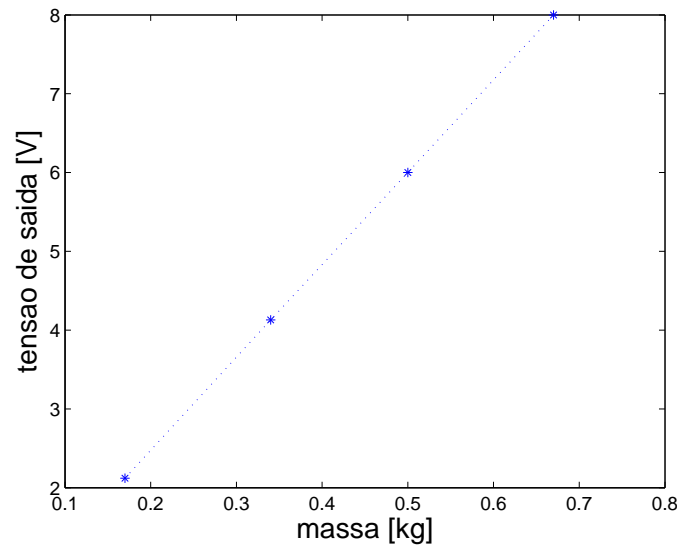
**Figura 3.28:** Vista explodida do dispositivo de medição da força magnética.

em uma balança de 0,22 g de precisão.

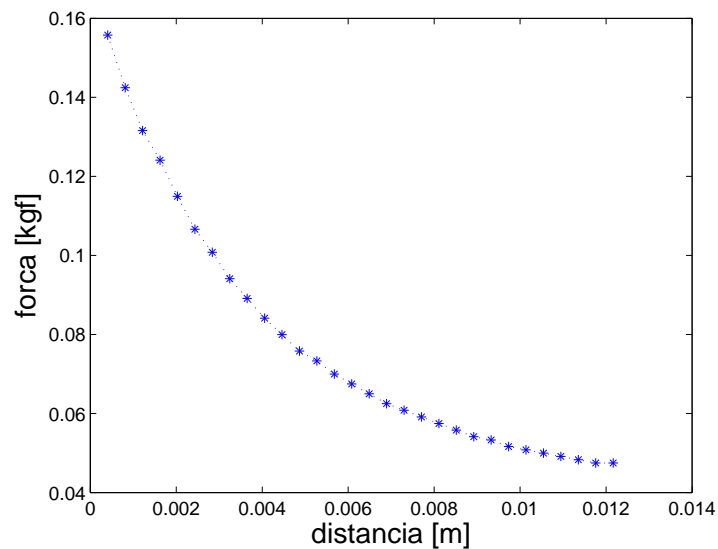
O sinal de saída da célula de carga foi amplificado e lido com um voltímetro portátil. A partir dos valores indicados na Figura (3.29) pode-se inferir que o ganho de amplificação do sistema de leitura é de 0,0833 kgf/V.

Utilizando-se o dispositivo, montaram-se duas bobinas, alimentadas com 30V. Mediu-se a força  $f$  e a corrente  $i$  em função da distância  $y$  entre o eletromagneto e a lâmina, dando origem ao gráfico da Figura (3.30).

O experimento foi repetido para lâminas com espessuras de 0,20 mm, 0,15 mm, e 0,10 mm, dando origem à Figura (3.31). Observa-se que a força de atração



**Figura 3.29:** Curva de calibração da célula de carga.

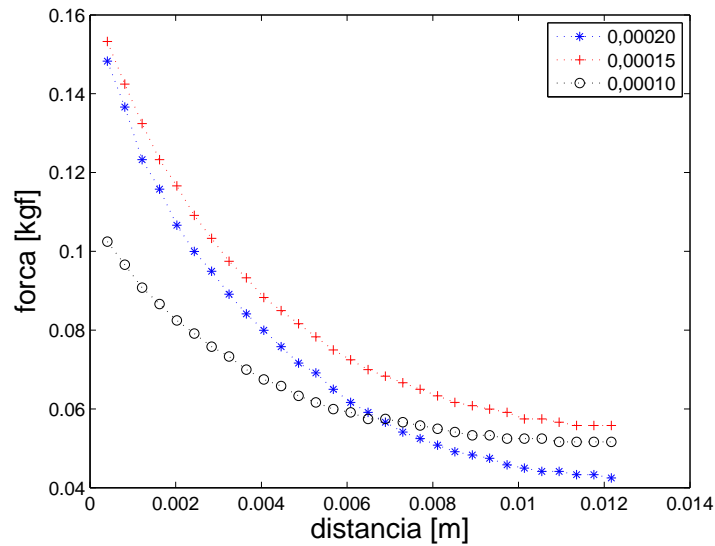


**Figura 3.30:** Curva de força eletromagnética versus distância da lâmina ao eletromagneto.

é maior para lâminas de maior espessura, provavelmente em função da maior massa de ferro da liga por unidade de comprimento da lâmina.

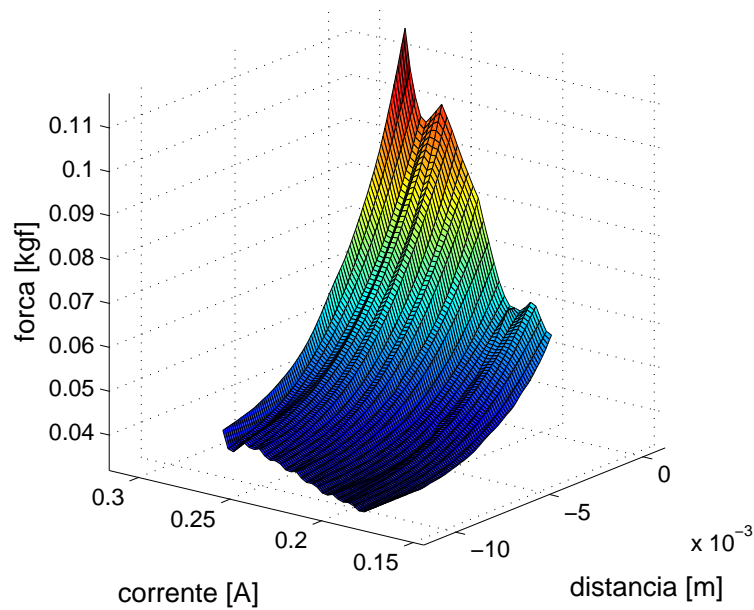
Este resultado pode ser empregado para alterar a resposta dinâmica da lâmina e melhorar da controlabilidade do movimento caso o atuador sature sem que a posição desejada seja alcançada. Em outras palavras, será preciso estimar, através de testes, que espessura de lâmina apresenta melhor desempenho dinâmico. O ganho na força de atração em função da espessura contrapõe-se ao aumento de rigidez da lâmina.

Para que a lâmina se deforme de modo controlado, de acordo a equação de deslocamento (2.3), os atuadores precisam ter sua força de atração magnética mo-



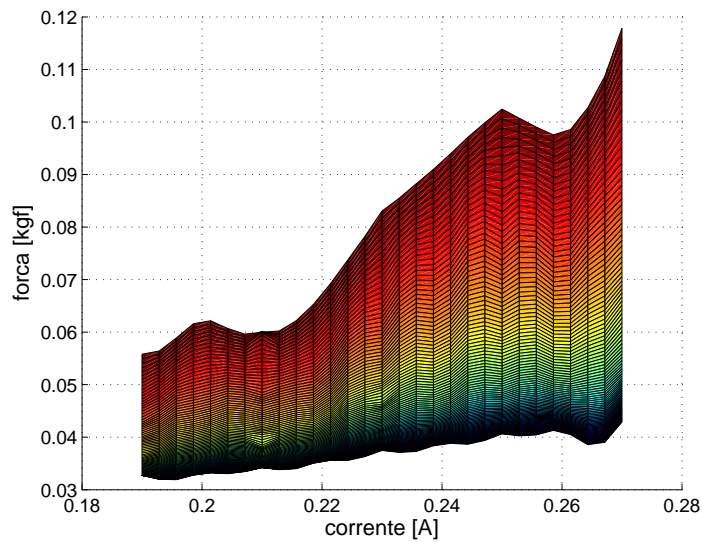
**Figura 3.31:** Força de atração magnética versus espessura da lâmina.

dulada por um sistema de controle. Portanto, repetiu-se o experimento variando-se a corrente, de modo a gerar os dados da Figura (3.32).

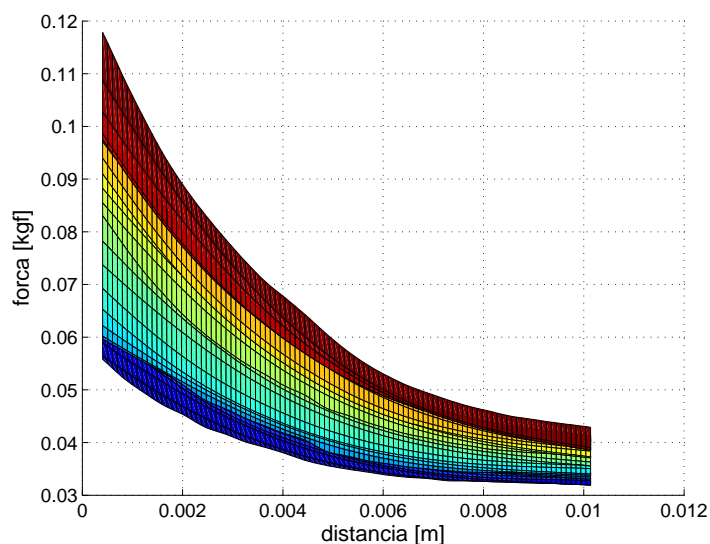


**Figura 3.32:** Força de atração magnética em função da distância e da corrente.

Observa-se na Figura (3.32) a relação entre força de atração magnética  $f_{mag}$  e a distância  $d$  entre o atuador e a lâmina. Observa-se também a relação entre a força de atração magnética  $f_{mag}$  e a corrente  $i$ . Este comportamento do atuador permite o controle da força  $f_{mag}$  em função da corrente  $i$  e da distância  $d$ , o que pode ser feito através de uma fonte de corrente com amplitude modulada por um sistema de controle.



**Figura 3.33:** Força de atração versus corrente para diversos valores de distância, as menores distâncias geram forças mais elevadas.



**Figura 3.34:** Força de atração versus distância para diversos valores de corrente, os maiores valores de corrente geram forças mais elevadas.

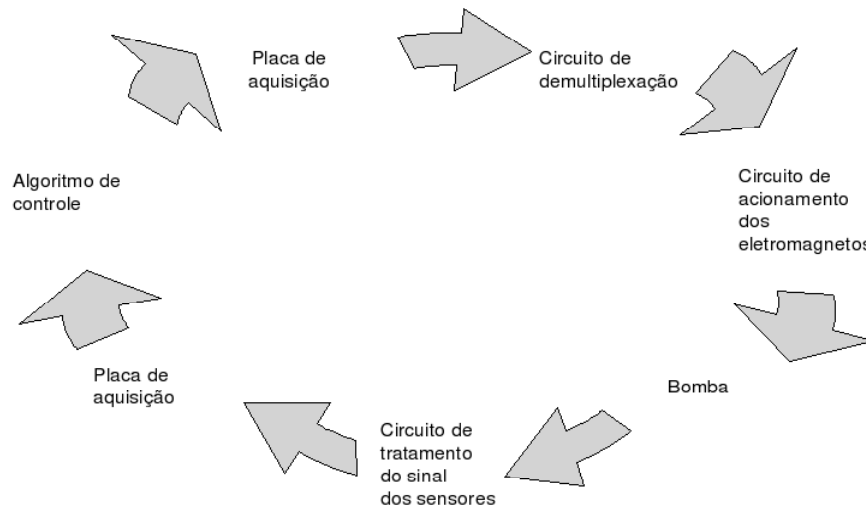
Através de uma equação que represente, de forma aproximada, o comportamento do eletromagneto, Figura (3.32), e da posição da lâmina, é possível determinar a corrente que gera a força determinada pelo sistema de controle.

### 3.2.3 Descrição da bomba ondulatória experimental

Nesta subsecção serão descritos os componentes de *hardware* que compõem o sistema de controle da lâmina.

### 3.2.3.1 Integração do *hardware* de controle

Os circuitos que compõem o *hardware* de controle do protótipo então integrados de acordo com a Figura (3.35) e seus componentes são descritos a seguir.



**Figura 3.35:** Esquema de integração do *hardware* de controle.

### 3.2.3.2 Interface de I/O

Foi utilizada uma interface de I/O modelo NI PCI-6229 de 250 ks/s, com 4 canais de saída analógicos de 16 bits de resolução, 32 canais de entrada analógicos e 48 bits que podem ser configurados como entrada ou saída digitais. Detalhes técnicos podem ser vistos em National Instruments (2006).

### 3.2.3.3 Circuito de acionamento dos eletromagnetos

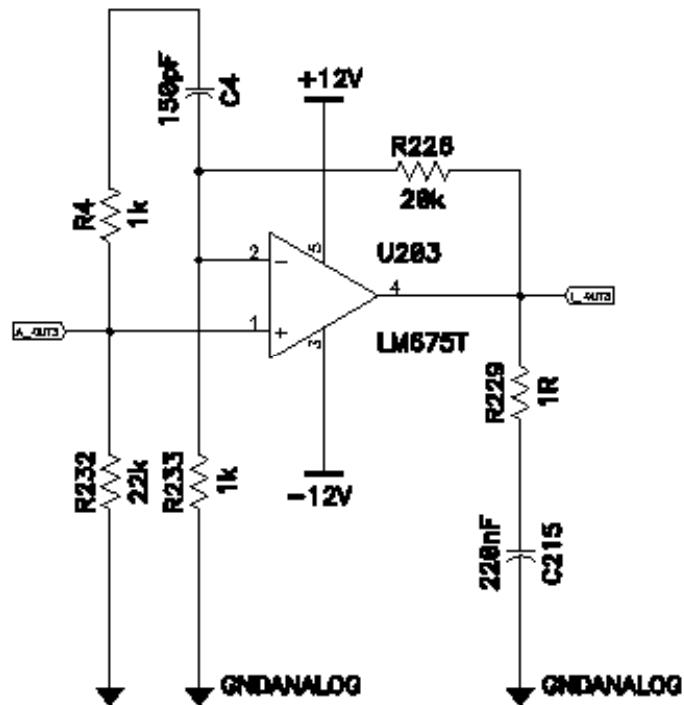
O algoritmo de controle determina a força eletromagnética necessária. Conhecida a força e a distância da lâmina, calcula-se a corrente que deve ser aplicada a cada eletromagneto. Conhecida a corrente, calcula-se a tensão necessária em cada fonte de corrente. Conhecida a tensão em cada fonte de corrente, gera-se um sinal de tensão utilizando os canais D/A da interface de I/O. Estes sinais de tensão acionam fontes de corrente que impõem corrente nos eletromagnetos, produzindo as forças desejadas. Utilizam chaves analógicas e circuitos mantenedores de tensão, *zero order hold*, que são utilizados para comandar 12 fontes de corrente através de 4 canais D/A.

Para distribuir o sinal analógico para os circuitos de acionamento foi preciso

criar um circuito de demultiplexação, o que permite endereçar o sinal dos quatro canais para os circuitos de acionamento dos atuadores. Foram empregados três endereços digitais para fazer o envio do sinal. Cada endereço habilita, por vez, o envio de quatro sinais analógicos.

Cada canal dispõe de um sub-circuito que armazena o último sinal analógico recebido pela placa de aquisição. Este valor é mantido até que seja recebido um novo valor, mantendo ativa a última ação de controle mesmo após sua desativação.

O circuito da fonte de corrente é formado por um amplificador operacional, ver Figura (3.36). A tensão imposta na entrada não-inversora determina a intensidade da corrente imposta ao eletromagneto. A Figura (3.37) mostra o comportamento linear do circuito de acionamento dos eletromagnetos, nela se verifica um ganho de corrente de 0.24 A/V. O circuito permite regular a corrente máxima com que as bobinas serão acionadas, de modo a proteger os eletromagnetos, garantindo um máximo de 400mA em regime para fios AWG 26.



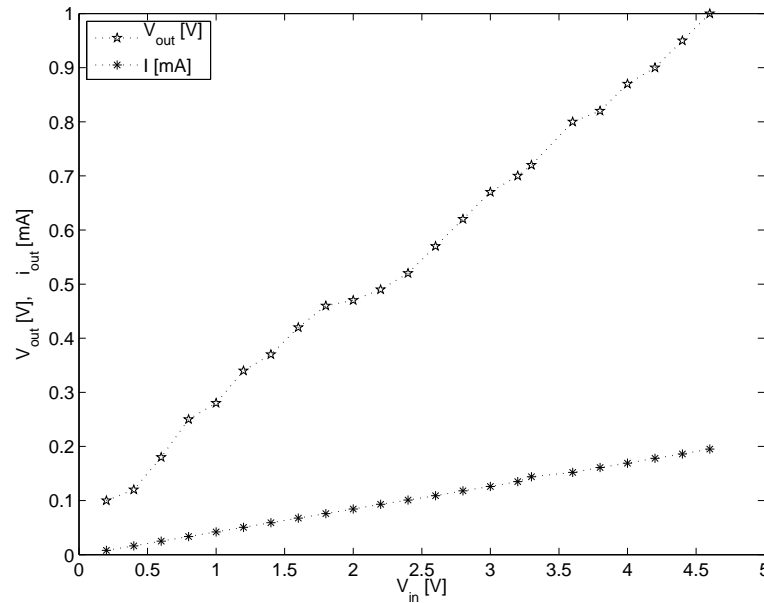
**Figura 3.36:** Circuito da fonte de corrente de acionamento dos eletromagnetos.

### 3.2.3.4 Circuito de tratamento do sinal dos sensores de posição

A posição da lâmina foi medida em cinco pontos. Cada posição correspondendo a um sinal de tensão medido com um circuito eletrônico que fez o tratamento do sinal analógico.

A disposição final do circuito de tratamento do sinal dos sensores diferenciais



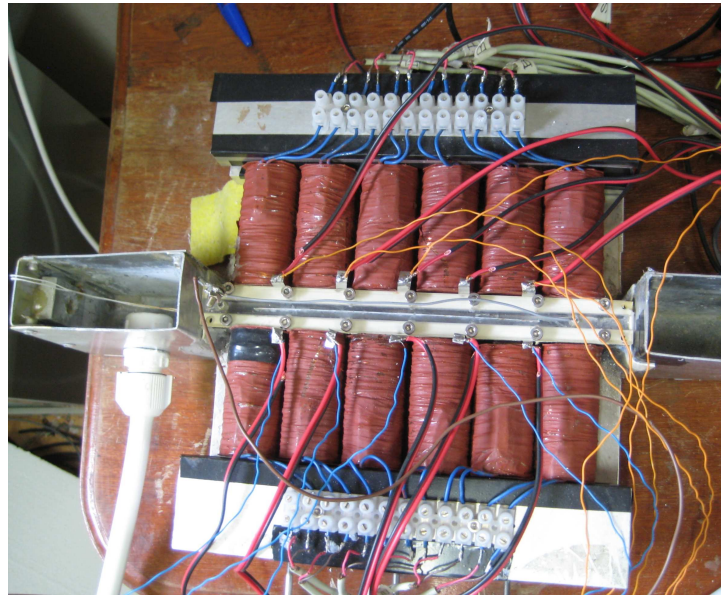


**Figura 3.37:** Comportamento linear do circuito de acionamento dos eletromagnetos.

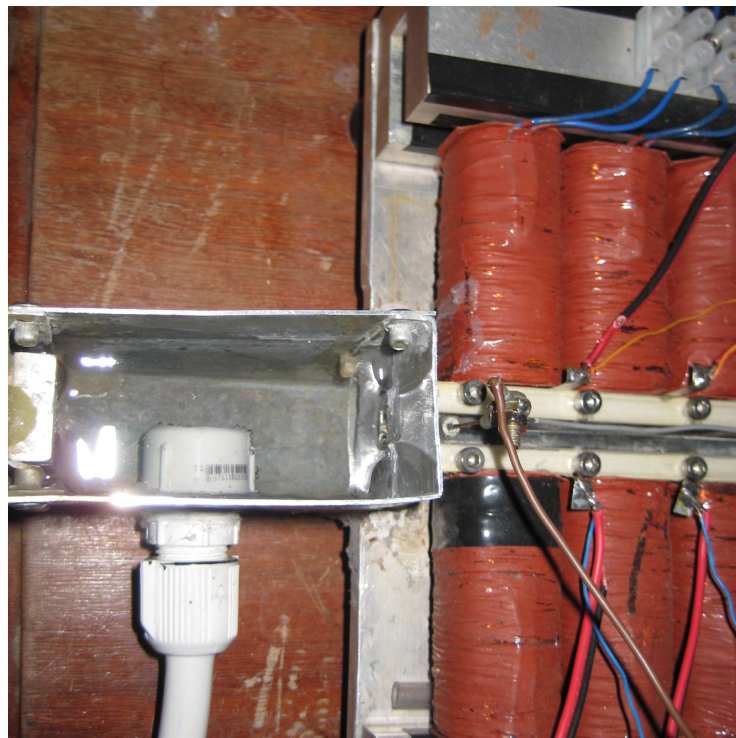
capacitivos pode ser vista na Figura (2.7). Uma onda senoidal é gerada com frequência e amplitudes moduladas em uma fonte de sinal. O sinal é filtrado, sendo dividido em dois ramos. Um ramo é levado a um seguidor, enquanto que o outro ramo é levado a um inversor, produzindo, desta forma duas ondas senoidais em oposição de fase. Ambos os sinais seguem para amplificadores, o que permite um ajuste individual. Na sequência os sinais excitam as placas dos capacitores diferenciais. Dois circuitos seguidores de alta impedância recebem o sinal, o retificam em uma ponte de diodos completa. Um filtro passa-baixa atenua os ruídos de alta frequência. Finalmente ambos os sinais se encontram na entrada de um somador. O sinal de saída foi lido por uma placa de aquisição.

### 3.2.3.5 Bombeamento da Bomba Experimental

Foram realizados testes de bombeamento nos quais se verificou que o escoamento na seção de descarga ocorre em golfadas em lados alternados na lâmina ondulatória, o que está de acordo com o previsto nos resultados de simulações numéricas, obtidos pela solução da equação de Navier-Stokes. Foi utilizado um corante branco como marcador de fluxo, colocado na seção de entrada da bomba e se registrou seu escoamento na seção de descarga da bomba. Verifica-se que a Figura (3.40)a) apresenta uma golfada em oposição de fase com a golfada apresentada na Figura (3.40)f). As Figuras (3.38) e (3.39) apresentam aproximação progressiva da imagem da seção de descarga da bomba experimental.



**Figura 3.38:** Vista superior do protótipo experimental.



**Figura 3.39:** Vista superior da seção de descarga do protótipo experimental.



a)



b)



c)



d)



e)



f)

**Figura 3.40:** Escoamento pulsátil na seção de descarga da bomba.

## 4 Comentários Finais

Foi desenvolvido um modelo numérico de bomba ondulatória empregando o método dos elementos finitos. Como a bomba ondulatória tem uma fronteira móvel no interior de seu domínio, foi empregado um processador de elementos finitos capaz de incorporar as condições que esta fronteira móvel impõe ao escoamento.

Para que o projeto de bomba ondulatória resultante pudesse ser ótimo em algum sentido, desenvolveu-se um índice de desempenho. O índice de desempenho escolhido estima a eficiência mecânica da bomba ondulatória. Equações e algoritmos para calcular a eficiência mecânica do bombeamento foram desenvolvidos. Outros índices podem ser propostos. Este índice foi aplicado em duas possíveis configurações de bomba, uma com a extremidade da lâmina na seção de saída fixa ao canal e outra com essa extremidade livre. Foi possível afirmar que a eficiência energética é superior na lâmina com a extremidade fixa.

Uma campanha de simulação numérica foi realizada para calcular a eficiência mecânica em função da frequência e da amplitude do movimento ondulatório. Esta campanha revelou que as frequências necessárias são muito baixas comparadas às frequências utilizadas em bombas de lâminas oscilatórias. Este fato indica que este tipo de bomba pode ser atraente em aplicações que requeiram baixas tensões de cisalhamento. As simulações numéricas permitem afirmar que a frequência de deslocamento transversal de 15 Hz juntamente com uma amplitude de deslocamento transversal de  $0,0025m$  deve gerar uma eficiência mecânica de cerca de 20%, um ganho de pressão de cerca de 0,3 mca e uma vazão de cerca de 0,1 kg/s para a lâmina com  $0,02m$  de comprimento e o canal com seção transversal de  $0,01 \times 0,01m$ .

As etapas de simulação numérica empregadas neste trabalho constituem uma proposta de metodologia para projetar bombas ondulatórias.

Um modelo experimental de bomba ondulatória foi construído com atuadores eletromagnéticos e sensores de posição capacitivos. Os atuadores e sensores foram ensaiados e apresentaram sinal de resposta de acordo com o previsto na literatura.

Resultados experimentais preliminares visam indicar a viabilidade da metodologia para projetar bombas ondulatórias.

É esperado que os resultados de ganho de pressão e de velocidade obtidos experimentalmente sejam menores que aqueles calculados por simulação numérica. Este fato decorre das diferenças entre modelo numérico e modelo experimental. A principal diferença entre o modelo numérico e o modelo experimental está na hipótese de escoamento em duas dimensões utilizada no modelo numérico. Esta modelagem do escoamento em duas dimensões não leva em conta o escoamento de fluido por debaixo e por encima da lâmina. Este escoamento certamente reduz o ganho de pressão, a velocidade e a eficiência mecânica.

Uma metodologia de projeto de bombas ondulatórias requer testes experimentais mais numerosos e abrangentes do que aqueles que puderam ser realizados neste trabalho. Os resultados obtidos apontam para a viabilidade da proposta de metodologia de projeto de bombas ondulatórias.

## 5 Conclusões

Foi proposta e implementada uma metodologia de projeto para bombas ondulatorias baseada em simulações de elementos finitos de escoamento viscoso e incompressível. As simulações numéricas permitem avaliar a eficiência mecânica.

Ao confinar o movimento ondulatorio em um canal, simulações numéricas indicam que este movimento efetivamente provoca bombeamento com velocidades transversais baixas.

Ao confinar o movimento ondulatorio em um canal, resultados experimentais indicam que este movimento efetivamente provoca bombeamento com baixas velocidades transversais.

Resultados numéricos indicam que existem condições em que bomba é mais eficiente do ponto de vista do trabalho mecânico. Na geometria adotada, a maior eficiência ocorreu a amplitudes de 2,5 mm, na frequência de 15 Hz e com as duas extremidades da lâmina sem deslocamento transversal.

Potencialmente, a disposição construtiva permite acionamento bidirecional, sendo, provavelmente, necessárias apenas modificações da direção de propagação da onda de deflexão transversal.

### 5.0.3.6 Trabalhos futuros

Desenvolver um modelo numérico em 3D e ampliar a gama de testes experimentais para poder ajustar o modelo numérico para torná-lo útil como ferramenta de projeto e otimização de bombas ondulatorias.

Melhorar o desempenho do *hardware*, eliminando o *offset* na placa de demultiplexação.

Substituir a lâmina por um atuador piezoelétrico, eliminando as bobinas eletromagnéticas e obtendo com isso uma otimização volumétrica do protótipo. Esta modificação permitirá colocar os capacitores pelo lado de fora do canal, eliminando os efeitos de corrosão das placas e simplificando seu isolamento.

Melhorar o algoritmo de controle com a introdução de um observador de estados.

## 6 APÊNDICE A - Fronteira móvel

### 6.1 Arquivos de definições da fronteira móvel

#### 6.1.1 Construção da malha

A malha inicial é definida nos arquivos *KL1.0* e em *parq2d.f*. O primeiro define o número de nós, número de elementos, número de nós por elemento e a quantidade de fronteiras fixas. O segundo define o contorno externo do volume de controle.

O arquivo *KL1.0* é constituído por um parágrafo de coordenadas nodais, seguido de um parágrafo de topologia e é finalizado com um parágrafo que define, por ordem crescente, a que fronteira pertence determinado nó.

Apesar da malha inicial ser grosseira, ela pode ser refinada diversas vezes, sendo o número de refinamento controlado no arquivo *cc2d.dat*, através do parâmetro NLMAX.

#### 6.1.2 Definição das condições de contorno

O modelo matemático que define a forma e o modo de deslocamento da fronteira móvel, apresentado na seção seguinte, é escrito nos arquivos *indat2d.f* e *bndry2d.f*. Em ambos os arquivos define-se a equação de deslocamento da onda, equação (6.3). No primeiro declaram-se as condições de contorno sobre o volume de controle. Nele a equação de deslocamento é usada para conhecer os nós aos quais se devem impor as condições de contorno. No segundo arquivo a equação de onda é utilizada para identificar os nós pertencentes à fronteira móvel.

No arquivo *indat2d.f* estão especificados a origem do sistema cartesiano que define a posição, comprimento e espessura da lâmina, a amplitude e a velocidade angular, bem como o parâmetro  $a$  que define o comprimento de onda da senoide. Também encontram-se definidos as componentes da velocidade, as pressões inici-



ais a jusante e a montante, o perfil de escoamento na seção de entrada, o número de seções de saída, suas posições, a posição da seção de entrada em estrito acordo com as dimensões pré-fixadas nos arquivos *parq2d.f* e *KL1.0*.

Pode-se ainda definir pontos nodais sobre os quais se quer avaliar a variação da pressão local ao longo de uma simulação.

### 6.1.3 Parâmetros da fronteira móvel

No arquivo *bdry.f* são novamente especificados a origem, o comprimento e espessura da lâmina, a amplitude e a velocidade angular, bem como o parâmetro  $a$  que define o comprimento de onda senoidal. Repete-se a definição da equação de deslocamento, equação (6.3), a primeira para a visualização e a segunda para a identificação dos nós pertencentes a fronteira móvel.

### 6.1.4 Parametrizações globais

No arquivo *cc2d.dat* encontram-se definidos os principais parâmetros da simulação, como a equação a ser resolvida, regime de escoamento, tipo de condições de contorno, a viscosidade cinemática do fluido de trabalho e especificações do método da solução numérica adotada.

Neste arquivo definem-se ainda os parâmetros para a solução da equação de Navier-Stokes (2.4), que incluem o tempo total da simulação e o instante inicial de integração. No caso em questão, o tempo de simulação adotado varia de 0.01s a 1.0 s, gerando uma solução a cada 0.01s,

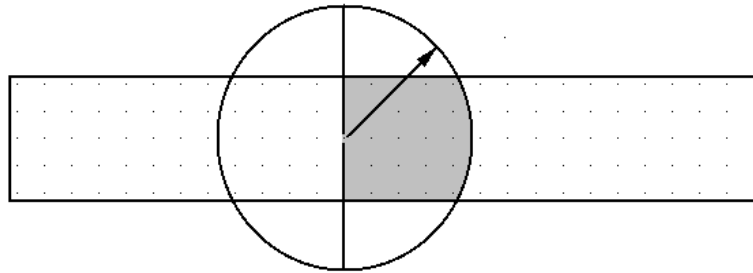
## 6.2 Região de alcance da fronteira móvel

A origem da fronteira é definida no ponto  $(X_{m1}, Y_{m1})$  pertencente a malha do volume de controle. Tomando a origem como referência, calcula-se para toda a malha, a distância  $ppp$ , conforme a equação (6.1). Ver Figura (6.1).

$$ppp = \sqrt{(XX - X_{m1})^2 + (YY - Y_{m1})^2} \quad (6.1)$$

A equação (6.1) define um círculo cuja origem coincide com a origem da localização da lâmina. Seu raio varia desde a origem até o comprimento da lâmina, sendo considerados apenas os nós cujas coordenadas  $(XX, YY)$  estiverem sobre o ponto  $(X_{m1}, Y_{m1})$  ou a sua direita, ou seja, excluem-se os nós da malha contidos

no semi-círculo a esquerda da origem.

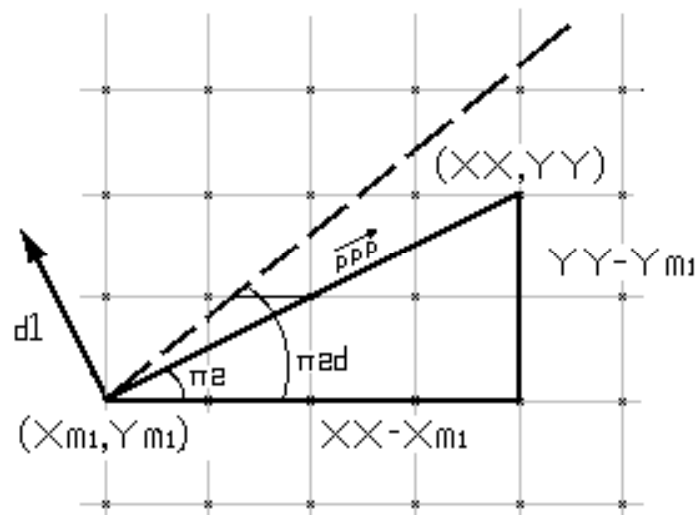


**Figura 6.1:** O retângulo representa a fronteira fixa e a região escura contém os nós que podem pertencer a Fronteira Móvel.

Dessa forma, fica definida a região de alcance da fronteira artificial móvel pela interseção do domínio da fronteira fixa com o domínio da equação (6.1).

### 6.3 Nós pertencentes à fronteira móvel

O Featflow precisa conhecer os nós que pertencem a fronteira móvel para poder impor os valores de velocidade e pressão sobre eles. Para isso é preciso continuar a restringir o domínio. As regras apresentadas a seguir efetuarão cálculos sobre as coordenadas nodais do domínio. Note-se que a identificação desses pontos deve ser feita a cada interação, de acordo com a equação (2.3).



**Figura 6.2:** Determinação dos nós pertencentes à fronteira móvel.

Para cada nó pertencente ao domínio da fronteira móvel, calcula-se o ângulo da coordenada  $(x, y)$  desse nó com a origem, ver equação (6.2).

$$\pi_2 = \arcsen \left( \frac{YY - Y_{m1}}{ppp} \right) \quad (6.2)$$

Neste ponto, recupera-se a função de deslocamento da fronteira, equação (2.3), que na sua forma discreta assume o formato da equação (6.3). Note-se que a variação espacial da função de deslocamento discreta é dada pelas coordenadas nodais da malha de elementos finitos empregada, ou seja, varia em função da densidade da malha utilizada.

$$\begin{aligned} \omega_f(x(i), t(n)) = A \left( 1 - \exp^{-(XX(i) - X_{m1})} \right) \left( 1 - \exp^{(XX(i) - X_{m1}) - 20} \right) \\ \text{sen} \left( \left( \frac{\pi}{a} \right) (XX(i) - X_{m1}) - \phi t_d \right) \end{aligned} \quad (6.3)$$

onde  $t_d = n\Delta t$ ,  $i$  é o nó pertencente ao comprimento da lâmina,  $n$  é o instante discreto no tempo, e  $t_d$  é o intervalo de tempo entre soluções consecutivas.

Relacionando as equações (6.1) e (6.3) é possível avaliar a posição angular da fronteira em relação a origem, equação (6.4).

$$\pi_{2d} = \arcsen \frac{\omega_f}{ppp} \quad (6.4)$$

A equação acima gera uma relação direta entre a variação espaço-temporal da fronteira móvel e a malha de nós. Ela permite ao algoritmo do Featflow ler a região do domínio sobre o qual atua a fronteira móvel e encontrar os nós que correspondem aos pontos da equação (6.3) que varia espacial e temporalmente, isso em função dos ângulos auxiliares, medidos em relação à origem. A seguir determina-se o ângulo auxiliar  $\pi_{1d}$ .

$$\pi_{1d} = \pi_2 - \pi_{2d} \quad (6.5)$$

## 6.4 Nós internos à fronteira móvel

Por fim, determina-se o valor de  $d_1$ , equação (6.6), que possui direção perpendicular a fronteira, ver Figura (6.2).

$$d_1 = ppp \text{sen}(\pi_{1d}) \quad (6.6)$$

Desta forma, qualquer ponto ao longo das abscissas, a partir da origem  $X_{m1}$ ,

com ordenada  $d_1$  em módulo, menor ou igual a espessura da lâmina, deve estar contido no domínio da fronteira móvel. O vetor  $d_1$  será sempre perpendicular a superfície da fronteira móvel para o nó considerado.

Os nós cujas coordenadas atendem a essa condição, terão pressão e velocidade anuladas. Dessa forma é imposto que não haja escoamento na fronteira móvel ou através dela, e portanto, a pressão deve ser nula para nesta região.

## 6.5 Componentes da velocidade da fronteira móvel

Apesar da velocidade de escoamento do fluido sobre a superfície da fronteira móvel ser nula, a região correspondente a fronteira móvel como um todo, por definição é móvel. Deve-se informar ao código componentes da velocidade  $\vec{u}$ , ou seja,  $v_x$  e  $v_y$ .

$$v_x = 0 \quad (6.7)$$

$$v_y = -\phi A (1 - \exp^{-(XX - X_{m1})}) (1 - \exp^{(XX - X_{m1}) - 20}) \cos\left(\frac{\pi}{a}(XX - X_{m1}) - \phi t_d\right) \quad (6.8)$$

O movimento representado pela equação (6.3) é o de uma senoide que se propaga no espaço, entretanto, não há deslocamento na direção  $x$ . Seus movimentos estão limitados a direção  $y$ , na qual se comporta como um oscilador harmônico simples, com amplitude e fase definidos ao longo da lâmina pela equação (6.8).

## 7 APÊNDICE B - Algoritmos em linguagem C

### 7.1 Cálculo do índice de desempenho

A seguir é descrito o algoritmo *func13.c* de cálculo do índice de desempenho para os arquivos soluções u.n.avx gerados pelo Featflow1.3RC3.

```

/*****
*****

```

Flávio José Aguiar Soares - Junho 2006

```

*****

```

```

/*****
*****

```

Este programa calcula o índice de desempenho da Bomba Numérica, para isto manipula as variáveis de saída dos arquivos solucoes do programa se simulação numerica de escoamentos Featflow1.3RC3, da seguinte forma:

- Le n arquivos de dados u.n.inp, para os quais:
- Le o numero de nohs na primeira linha do programa;
- Le o paragrafo das coordenadas nodais, onde salva o numero de nohs e as coordenadas X e Y;
- Le, para cada noh, suas propriedades físicas relacionadas de pressão e velocidade;
- Le dois arquivos auxiliares contendo os nohs das secões de entrada e saída da malha de solucoes;
- Calcula, identifica e le os nohs na fronteira móvel;
- Calcula o trabalho na entrada e na saída da malha de solucoes;

- Calcula o trabalho ao longo da fronteira móvel;
- Calcula o rendimento para cada arquivo e o imprime num arquivo de dados específico.
- Calcula a diferença de pressão entre entrada e saída e a vazão para cada arquivo e a imprime num arquivo de dados.

```
*****
*****/
```

```
#include <stdio.h> /* includes de I/O */
#include <stdlib.h>
#include <string.h>
#include <math.h> /* includes de funcoes matemáticas */
#include "nrutil.h" /* includes do 'numerical recipes' */
#define pi 3.1415926535897931

int main()
{
    char line[100];
    unsigned int ang=6;
    unsigned int num_node_section=0;
    unsigned int i=0,j=0,k=0,m=0,n=0,o=0,q=0,r=0,s=0,t=0;
    unsigned int num_files,num_node;
    unsigned int node_coor[9500];
    unsigned int node_coor_next[9500];
    unsigned int node_in[65],node_out[65];
    unsigned int node_fm_up[165],node_fm_down[165];
    unsigned int node_fm_up_dx[165],node_fm_down_dx[165];
    unsigned int xm1d,ym1d; /* [cm] */
    float xm1=2.000,ym1=0.500; /* [cm] */
    float dy=0.03125,dx=0.125,de=2*dy; /* [cm] */
    float rho=1.0; /* [g/cm^3] */
    float blade_rho=7.9; /* [g/cm^3] */
    float tf=0.01; /* [s] */
    float damp=0,dfase=0,deps=0.02,dlarge=1.0;
    float beam=20.0,beam_length=0; /* [cm] */
    float area_in = (dy*dlarge); /* [cm^2] */
    float area_out = (dy*dlarge); /* [cm^2] */
    double elast = 210.0e5; /* [N/cm^2] */
    double inertia = dx*pow(deps,2)/3; /* [cm^3] */
    double n_fi[165],n_fi_next[165];
    double factor=0.000101972; /* [Pa] -> [mca] */
    double ni[200];
    double ddy[165]; /* [cm/s^2] */
    double y_fi[165],y_fi_next[165],y_fi_next2[165];
    double xi[165],xi_n[165];
    double x_fm_up[165],y_fm_up[165];
    double x_fm_up_dx[165],y_fm_up_dx[165];
```

```

double x_fm_down[165],y_fm_down[165];
double x_fm_down_dx[165],y_fm_down_dx[165];
double coorx[9500],coory[9500];
double coorx_next[9500],coory_next[9500];
double press[9500],velx[9500];
double press_next[9500],velx_next[9500];
double press_in[65],press_out[65]; /* [cm^2/s^2] */
double press_med_up[165],press_med_down[165];
double velx_in[65],velx_out[65]; /* [cm/s] */
double vazão[65]; /* [kg/s] */
double vaux11[165],vaux12[165];
double vaux6,vaux7;
double vaux13,vaux14,vaux15;
double vaux20,vaux21;
double vaux29;
double vaux30[165],vaux31[165],vaux32[165];
double vaux33[165],vaux34[165],vaux35[165];
double vaux36[165],vaux37[165],vaux38[165];
double vaux39[165],vaux40[165],vaux41[165];
double vaux42[165],vaux43[165],vaux44[165],vaux45[165];
double force,force_in,force_out; /* [N] */
double work,work_in,work_out; /* [Nm] */
double delta_x,delta_m;
double delta_y[165],delta_press[165],delta_vel[165];
double fcort_p[165],fcort_n[165];

char *aux,*aux1,*aux2,*aux3,*aux4,*aux6,*aux7,*aux8;
char *file_now;
char *line1,*line2,*line3,*line4;

FILE *fileavs;
FILE *write_ni,*write_delta_press,*write_flow_out;
FILE *read_file_in,*read_file_out;

/* definição dos parametros da equação de forma */
printf("Entre com o valor da Amplitude (0<->0.5), DAMP.\n\n");
printf("valores padrões 0.10, 0.15, 0.20, 0.25, 0.30\n\n");
scanf("%f",&damp);

printf("Entre com o valor da freq. de deslocamento da onda.\n");
printf("valores padrões:\n");
printf("1 = 1Hz, 5 = 5Hz, 10 = 10Hz, 15 = 15Hz, 20 = 20Hz\n\n");
scanf("%f",&dfase);
dfase=dfase*2*pi;

/* printf("Finalmente, entre com o numero dos \n"); */
/* printf("'n' arquivos u.n.inp, a serem lidos \n"); */
/* scanf("%d",&num_files); */
num_files = 100;

/* geração da advertência */

```

```

system("play startup3.wav");
printf("\n\n Este programa farah a leitura \n");
printf("\n dos %d arquivos \n\n",num_files);

line1=calloc(100,sizeof(char));
line2=calloc(100,sizeof(char));
line3=calloc(100,sizeof(char));
strcat(line1,"pertencentes ao directorio ");
sprintf(line2,"a");
sprintf(line3,"%f",damp*100);
strcat(line2,line3);
strcat(line2,"_w");
sprintf(line3,"%f",dfase/(2*pi));
strcat(line2,line3);
strcat(line2,"_avs \n\n");
strcat(line1,line2);
printf("%s",line1);
sprintf(line3,".txt");
strcat(line2,line3);
free(line1);
free(line3);

vaux6 = damp;
vaux7 = dfase;

/* abre arquivo de escrita das pressoes medias      */
write_ni=fopen("rendimento.txt","w");
write_delta_press=fopen("pressão_out.txt","w");
write_flow_out=fopen("vazão_out.txt","w");

/* ROTINA DE LEITUNA DOS n ARQUIVOS      */
/* Rotina de atualizacao do nome do arquivo a ser lido      */
/* aloca memoria para a string nome do arquivo 'file_now'      */
/* nome de file_now inicia com 'u.'      */
/* seguido de um index inteiro n, adicionado a variavel 'aux'*/
/* que adicionado a file_now fica 'u.n', que eh finalizado com */
/* a adicao de '.inp', ficando 'u.n.inp', onde n eh o numero de */
/* arquivos a ser lido, a cada interacao, n eh incrementado */
for (n=1;n<=num_files;n++)
{
    /* Rotina de leitura do numeros dos nohs na secao de entrada */
    /* Os valores lidos serao salvos no vetor node_in[m]      */
    read_file_in=fopen("node_in.txt","r");
    while (!feof(read_file_in)) {
        fscanf(read_file_in,"%s",&line);
        node_in[m]=atoi(line);
        /* printf("m=%d,node_in[m]=%d\n",m,node_in[m]);      */
        m++;
    }
    fclose(read_file_in);

    num_node_section=m;

```



```

/*Rotina identica a anterior, correspondente aos nos de saida,*/
/* Neste caso, salvos em node_out[1] */
/* Salva o valor do numero de nodos das secoes de entrada/saida*/
read_file_out=fopen("node_out.txt","r");
while (!feof(read_file_out)) {
fscanf(read_file_out,"%s",&line);
node_out[o]=atoi(line);
/* printf("o=%d,node_out[o]=%d\n",o,node_out[o]); */
o++; }
fclose(read_file_out);
o=0;

/* Geracao do nome do arquivo n+1 a ser aberto */
file_now=calloc(15,sizeof(char));
aux=calloc(4,sizeof(char));
strcat(file_now,"u.");
sprintf(aux,"%d",n); /* sprintf(aux,"%d",n+1); */
strcat(file_now,aux);
strcat(file_now,".inp");
free(aux);

/* Abertura do arquivo com a variavel de interesse */
/* Leitura do numero de nos na primeira linha do programa */
fileavs=fopen(file_now,"r");
if(fgets(line,100,fileavs))
{
aux1=calloc(4,sizeof(char));
aux8=calloc(100,sizeof(char));
for (j=0;j<=5;j++) {
sprintf(aux1,"%c",line[j]);
strcat(aux8,aux1); }
num_node=atoi(aux8);
}
free(aux8);
free(aux1);
/* Leitura dos nos e das coordenadas nodais */
/* Enquanto nao chega ao fim do arquivo */
while (!(num_node==i))
{
if(fgets(line,100,fileavs))
{
aux1=calloc(4,sizeof(char));
aux2=calloc(8,sizeof(char));
for (j=0;j<=7;j++) {
sprintf(aux1,"%c",line[j]);
strcat(aux2,aux1); }
node_coor[i]=atoi(aux2);
node_coor_next[i]=node_coor[i];
free(aux1);
free(aux2);
}
}
}

```

```

/* coordenadas nodais nao se alteram          */

    /*printf("i = %d, node_coor = %d\n",i,node_coor[i]);    */

    /* Leitura do paragrafo das coordenadas nodais em X e em Y */
        aux1=calloc(4,sizeof(char));
        aux3=calloc(100,sizeof(char));
for (j=9;j<=22;j++) {
sprintf(aux1,"%c",line[j]);
strcat(aux3,aux1); }
coorx[i]=atof(aux3);
coorx_next[i]=coorx[i];
        free(aux1);
        free(aux3);

        aux1=calloc(4,sizeof(char));
        aux4=calloc(100,sizeof(char));
for (j=24;j<=37;++j) {
sprintf(aux1,"%c",line[j]);
strcat(aux4,aux1); }
coory[i]=atof(aux4);
coory_next[i]=coory[i];
        free(aux1);
        free(aux4);

    /* printf("i=%d, coorx=%f, coory=%f\n",i,coorx[i],coory[i]); */
i++;
}
}
i=1;

/* Leitura das variaveis da malha do arquivo n+1          */
while (!(feof(fileavs)))
{
fscanf(fileavs,"%s",&line);
if (!strcmp(line,"stream,m/s"))
{
while (!(feof(fileavs)))
{
if(fgets(line,100,fileavs))
{
aux1=calloc(4,sizeof(char));
aux6=calloc(100,sizeof(char));

for (k=8;k<=22;k++) {
sprintf(aux1,"%c",line[k]);
strcat(aux6,aux1); }
velx[t]=atof(aux6);

        free(aux1);
        free(aux6);
/* velocidade Vx          */

```

```

        aux1=calloc(4,sizeof(char));
        aux7=calloc(100,sizeof(char));
    for (k=53;k<=67;++k) {
    sprintf(aux1,"%c",line[k]);
    strcat(aux7,aux1); }
    press[t]=atof(aux7);
        free(aux1);
        free(aux7);
        /* e pressao */

    /* printf("t=%d, velx=%f, press=%f\n",t,velx[t],press[t]);*/

t++;
}
}
    }
}
fclose(fileavs);
free(file_now);
t=1;

/* Inicio dos calculos */
/* Leitura das variaveis nas secoes de entrada e saida */
/* Variacao de 0 a 32 */
for (s=0; s<num_node_section; s++)
{
    velx_in[s] = velx[node_in[s]];
    press_in[s] = press[node_in[s]];

    velx_out[s] = velx[node_out[s]];
    press_out[s] = press[node_out[s]];
    vazao[s] = velx_out[s]*area_out;

    /* pressao, entrada [cm^2/s^2] -> [mca] saida */
    /* P[Pa] = p/rho (turek) => multiplicar por rho */
    /* [cm^2/s^2]*[g/cm^3] = [g/(cm.s^2)]; Pa=kg/(m.s^2) */
    /* P[Pa] * factor -> P[mca] */
    /* vaux14 = (press_out[s]- press_in[s]) + vaux14; */
    vaux14 = press_out[s] + vaux14;
    /* vazao de saida=[cm/s]*[cm^2]*[g/cm^3]*0.001= [kg/s] */
    vaux15 = vazao[s]* rho * 0.001 + vaux15;
}
vaux14 = (vaux14/num_node_section)*rho*0.1*factor;

/* Calcula numero de colunas ocupadas pela lamina */
beam_length = ceil(beam/dx);

/* Discretizacao das coordenadas da origem da lamina */
xm1d = ceil(xm1/dx)-1;
ym1d = ceil(ym1/dy)-2;

```

```

/* Calculos ao longo da lamina, variacao de 0 a 160      */
for (q=0; q<= beam_length - 1; q++)
{
  xi[q] = q*dx;
  xi_n[q] = (q+1)*dx;

  /* Determinacao da posicao dos q-essimos elementos      */
  /* em 3 instantes consecutivos                          */

  /* y_fi1(i)=damp*(1-exp(-xi_[q]))*sin((pi/ang)         */
  /* (x_i(i))-dfase*tf*n); */
  /*.....ponta s solta.... */
  /* y_fi[q]= ym1 +vaux6*(1-exp(-xi[q]))                */
  /* sin((pi/ang)*(xi[q]) - vaux7*tf*n); */
  /* y_fi_next[q]=ym1+vaux6*(1-exp(-xi[q]))             */
  /* sin((pi/ang)*(xi[q])-vaux7*tf*(n+1)); */
  /* y_fi_next2[q]=ym1+vaux6*(1-exp(-xi[q]))            */
  /* sin((pi/ang)*(xi[q])-vaux7*tf*(n+2)); */

  /* y_fi1(i)= damp*(1-exp(-x_i(i)))*(1-exp(x_i(i)-20)) */
  /* sin((pi/ang)*(x_i(i))-dfase*tf*n); */
  /*.....ponta fixa..... */
  y_fi[q]= ym1 + vaux6*(1-exp(-xi[q]))*(1-exp(xi[q]-20))
  /* sin((pi/ang)*(xi[q]) - vaux7*tf*n);
  y_fi_next[q]=ym1+vaux6*(1-exp(-xi[q]))*(1-exp(xi[q]-20))
  /* sin((pi/ang)*(xi[q])-vaux7*tf*(n+1));
  y_fi_next2[q]=ym1+vaux6*(1-exp(-xi[q]))*(1-exp(xi[q]-20))
  /* sin((pi/ang)*(xi[q])-vaux7*tf*(n+2));

  /* Calculo da aceleracao para cada q-essimo elemento */
  delta_y[q] = y_fi_next[q] - y_fi[q]; /* [cm] */
  ddy[q] = (((y_fi_next2[q] - y_fi_next[q])/(tf))
  - ((y_fi_next[q]- y_fi[q])/(tf)))/tf; /* [cm/s^2] */

  n_fi[q] = ceil(y_fi[q]/dy);
  n_fi_next[q] = ceil(y_fi_next[q]/dy);

  printf("n=%d, q=%d, y_fi[q]=%f, n_fi[q]=%f\n",
  n,q,y_fi[q],n_fi[q]);
  printf("y_fi_next[q]=%f, n_fi_next[q]=%f\n",
  y_fi_next[q],n_fi_next[q]);

  /* Determinacao das coordenadas dos nohs */
  /* vizinhos a fronteira movel */
  y_fm_up[q] = n_fi[q]*dy + de;
  x_fm_up[q] = (xm1d + q)*dx;
  /* printf("x_fm_up[q] = %f\n",y_fm_up[q]); */
  /* printf("y_fm_up[q] = %f\n",y_fm_up[q]); */

```

```

    y_fm_down[q] = n_fi[q]*dy - de;
    x_fm_down[q] = (xm1d + q)*dx;
/* printf("x_fm_down[q] = %f\n",x_fm_down[q]); */
/* printf("y_fm_down[q] = %f\n",y_fm_down[q]); */

    y_fm_up_dx[q] = y_fm_up[q];
    x_fm_up_dx[q] = x_fm_up[q] + dx;
/* printf("x_fm_up_dx[q] = %f\n",x_fm_up_dx[q]); */
/* printf("y_fm_up_dx[q] = %f\n",y_fm_up_dx[q]); */

    y_fm_down_dx[q] = y_fm_down[q];
    x_fm_down_dx[q] = x_fm_down[q] + dx;
/* printf("x_fm_down_dx[q] = %f\n",x_fm_down_dx[q]); */
/* printf("y_fm_down_dx[q] = %f\n",y_fm_down_dx[q]); */

    /* Procura na matriz de coordenadas nodais */
    /* o noh correspondente as coordenadas escolhidas*/
    for (r=2;r<=num_node-1;r++)
        {
if (x_fm_up[q]==coorx[r] && y_fm_up[q]==coory[r])
    {
        node_fm_up[q] = node_coor[r];
        printf("n=%d,q =%d, node_fm_up[q] = %d\n",
            n,q,node_fm_up[q]);
    }
if (x_fm_down[q]==coorx[r] && y_fm_down[q]==coory[r])
    {
        node_fm_down[q] = node_coor[r];
        printf("n=%d,q =%d, node_fm_down[q] = %d\n",
            n,q,node_fm_down[q]);
    }
if (x_fm_up_dx[q]==coorx[r] && y_fm_up_dx[q]==coory[r])
    {
        node_fm_up_dx[q] = node_coor[r];
        printf("n=%d,q =%d, node_fm_up_dx[q] = %d\n",
            n,q,node_fm_up_dx[q]);
    }
        if (x_fm_down_dx[q]==coorx[r] && y_fm_down_dx[q]==coory[r])
    {
        node_fm_down_dx[q] = node_coor[r];
        printf("n=%d,q =%d, node_fm_down_dx[q]=%d\n",
            n,q,node_fm_down_dx[q]);
    }
        }

    /* Calculo do esforco cortante para o q_essimo elemento */
    /* .....ponta solta.....*/
    /* y_fi1[q] = damp*(1-exp(-xi_[q]))*sin((pi/ang)
    *(x_i[q])-dfase*tf*n); */
    /* vaux29=(pi/ang); */
    /* vaux30[q] = ((pi/ang)*(xi[q])-vaux7*tf*n); */

```

```

/* vaux31[q] = vaux6*exp(-(xi[q]))*sin(vaux30[q]);      */
/* vaux32[q] = -3*exp(-(xi[q]))*vaux6
*cos(vaux30[q])*vaux29;                               */
/* vaux33[q] = -3*exp(-(xi[q]))*vaux6
*sin(vaux30[q])*pow(vaux29,2);                       */
/* vaux34[q] = -(1-exp(-(xi[q])))*vaux6
*cos(vaux30[q])*pow(vaux29,3);                       */
/* fcort_n[q] = (vaux31[q]+vaux32[q]+vaux33[q]
+vaux34[q])*elast*inertia; */

/* Calculo do esforco cortante para o q_essimo+1 elemento */
/* vaux35[q] = ((pi/ang)*(xi_n[q])-vaux7*tf*n);        */
/* vaux36[q] = vaux6*exp(-(xi_n[q]))*sin(vaux35[q]);  */
/* vaux37[q] = -3*exp(-(xi_n[q]))*vaux6
*cos(vaux35[q])*vaux29;                               */
/* vaux38[q] = -3*exp(-(xi_n[q]))*vaux6
*sin(vaux35[q])*pow(vaux29,2);                       */
/* vaux39[q] = -(1-exp(-(xi_n[q])))*vaux6
*cos(vaux35[q])*pow(vaux29,3);                       */
/* fcort_p[q] = (vaux36[q]+vaux37[q]+vaux38[q]
+vaux39[q])*elast*inertia; */

/* Calculo do esforco cortante para o q_essimo elemento */
/*.....ponta fixa.....*/
/* y_fi1(i)= damp*(1-exp(-x_i(i)))*(1-exp(x_i(i)-20))
*sin((pi/ang)*(x_i(i))-dfase*tf*n); */

vaux29 = (pi/ang);
vaux30[q] = damp*exp(-xi[q])*(1-exp(xi_n[q]-20))
*sin((vaux29)*(xi[q])-dfase*tf*n);
vaux31[q] = -3*damp*exp(-xi[q])*(1-exp(xi_n[q]-20))
*cos((vaux29)*(xi[q])-dfase*tf*n)*vaux29;
vaux32[q] = -6*damp*exp(-xi[q])*exp(xi_n[q]-20)
*cos((vaux29)*(xi[q])-dfase*tf*n)*vaux29;
vaux33[q] = -3*damp*exp(-xi[q])*(1-exp(xi_n[q]-20))
*sin((vaux29)*(xi[q])-dfase*tf*n)*vaux29*vaux29;
vaux34[q] = -damp*(1-exp(-xi[q]))*exp(xi_n[q]-20)
*sin((vaux29)*(xi[q])-dfase*tf*n);
vaux35[q] = -3*damp*(1-exp(-xi[q]))*exp(xi_n[q]-20)
*cos((vaux29)*(xi[q])-dfase*tf*n)*vaux29;
vaux36[q] = 3*damp*(1-exp(-xi[q]))*exp(xi_n[q]-20)
*sin((vaux29)*(xi[q])-dfase*tf*n)*vaux29*vaux29;
vaux37[q] = -damp*(1-exp(-xi[q]))*(1-exp(xi_n[q]-20))
*cos((vaux29)*(xi[q])-dfase*tf*n)*vaux29*vaux29*vaux29;

fcort_n[q]=(vaux30[q]+vaux31[q]+vaux32[q]+vaux33[q]
+vaux34[q]+vaux35[q]+vaux36[q]+vaux37[q])*elast*inertia;

/* Calculo do esforco cortante para o q_essimo+1 elemento */
vaux38[q] = damp*exp(-xi_n[q])*(1-exp(xi_n[q]-20))
*sin((vaux29)*(xi_n[q])-dfase*tf*n);

```

```

vaux39[q] = -3*damp*exp(-xi_n[q])*(1-exp(xi_n[q]-20))
*cos((vaux29)*(xi_n[q])-dfase*tf*n)*vaux29;
vaux40[q] = -6*damp*exp(-xi_n[q])*exp(xi_n[q]-20)
*cos((vaux29)*(xi_n[q])-dfase*tf*n)*vaux29;
vaux41[q] = -3*damp*exp(-xi_n[q])*(1-exp(xi_n[q]-20))
*sin((vaux29)*(xi_n[q])-dfase*tf*n)*vaux29*vaux29;
vaux42[q] = -damp*(1-exp(-xi_n[q]))*exp(xi_n[q]-20)
*sin((vaux29)*(xi_n[q])-dfase*tf*n);
vaux43[q] = -3*damp*(1-exp(-xi_n[q]))
*exp(xi_n[q]-20)*cos((vaux29)*(xi_n[q])-dfase*tf*n)*vaux29;
vaux44[q] = 3*damp*(1-exp(-xi_n[q])) segue->
*exp(xi_n[q]-20)*sin((vaux29)*(xi_n[q])-dfase*tf*n)
*vaux29*vaux29;
vaux45[q] = -damp*(1-exp(-xi_n[q]))*(1-exp(xi_n[q]-20))
*cos((vaux29)*(xi_n[q])-dfase*tf*n)*vaux29*vaux29*vaux29;

fcort_n[q]= (vaux38[q]+vaux39[q]+vaux40[q]+vaux41[q]
+vaux42[q]+vaux43[q]+vaux44[q]+vaux45[q])*elast*inertia;

/*Agora que se conhece os nos da fm, pode-se determinar*/
/* o valor da variavel de interesse naquele noh */
press_med_up[q] = (press[node_fm_up[q]]
+ press[node_fm_up_dx[q]])/2;
press_med_down[q] = (press[node_fm_down[q]]
+ press[node_fm_down_dx[q]])/2;

vaux11[q] = press_med_up[q]*area_in*rho;/* [g.cm/s^2] */
vaux12[q] = press_med_down[q] * area_out * rho;
}

/* Somatoria dos trabalhos nas secoes de entrada e saida */
for (s=0; s<num_node_section; s++) {
    vaux20 = press_in[s] * area_in * velx_in[s]* tf + vaux20;
    vaux21 = press_out[s] * area_out * velx_out[s]*tf +vaux21;
}
work_in = vaux20*rho; /* [g.cm^2/s^2] */
work_out = vaux21*rho; /* [g.cm^2/s^2] */

delta_m = blade_rho * dx * deps * dlarge; /* [g] */
printf("delta_m=%f\n",delta_m);

/* Conhecidos os valores das variaveis em torno da fm */
/* pode-se determinar para cada q elemento da viga, o */
/* o q-essimo trabalho, e assim calcular a */
/* somatoria dos trabalhos dos q-essimos */
/* elementos da lamina */
for (q=0;q<= beam_length-1;q++) {/* forca em [g.cm/s^2] */
    force=fcort_p[q]-fcort_n[q]+vaux11[q]
-vaux12[q]+delta_m*ddy[q];
    work = fabs(force*delta_y[q]) + work; /*[g.cm^2/s^2] */
}

```

```

/* Convertendo para unidades do SI */
work_in = work_in*1e-7; /* [kg.m/s^2]*[m] = [N]*[m] */
work_out = work_out*1e-7; /* [kg.m/s^2]*[m] = [N]*[m] */
work = work*1e-7; /* [kg.m/s^2]*[m] = [N]*[m] */

/* RENDIMENTO TOTAL p/ o arq. n,
referente ao instante [t] */
ni[n] = fabs(work_out)/(fabs(work_in) + work);

printf("\nn = %d, ni[n] = %.10f\n\n",n,ni[n]);
printf("work_in=%.10f, work_out=%.10f, work=%.10f\n",
work_in,work_out,work);

/* Saida de resultados para impressao em arquivos */
/* Rendimento variando entre 0 <-> 1 */
/* Variacao de pressao entre entrada e saida em [mca] */
/* Vazão massica [kg/s] */
fprintf(write_ni,"%f\n ",ni[n]);
fprintf(write_delta_press,"%f\n ",vaux14);
fprintf(write_flow_out,"%f\n ",vaux15);

printf("terminei calculos do ciclo n = %d,
vou zerar variaveis\n",n);

/* variaveis auxiliares devem ser zeradas */

i = 0; j = 0; k = 0; m = 0; o = 0;
q = 0; r = 0; s=0; t=0;
vaux13=0; vaux14=0; vaux15=0; vaux20=0; vaux21=0;
delta_x=0; delta_m=0; force=0; force_in=0; force_out=0;
work =0; work_in=0; work_out=0;

for (i=0;i<=num_node_section;i++){
press_in[i]=0; press_out[i]=0;
velx_in[i]=0; velx_out[i]=0;
node_in[i]=0; node_out[i]=0; vazao[i]=0;
}

for (i=0;i<=num_node;i++){
press[i]=0; velx[i]=0;
press_next[i]=0; velx_next[i]=0;
}

for (i=1;i<=beam_length - 1;i++) {
xi_n[i]=0; /* xi_n = xi_next */
xi[i]=0; y_fi[i]=0;
x_fm_up[i]=0; y_fm_up[i]=0;
x_fm_up_dx[i]=0; y_fm_up_dx[i]=0;
x_fm_down[i]=0; y_fm_down[i]=0;
x_fm_down_dx[i]=0; y_fm_down_dx[i]=0;
}

```



```

y_fi_next[i]=0;          y_fi_next2[i]=0;
node_fm_up[i]=0;        node_fm_down[i]=0;
node_fm_up_dx[i]=0;    node_fm_down_dx[i]=0;
delta_vel[i]=0;        delta_press[i]=0;
delta_y[i]=0;          ddy[i]=0;
n_fi[i]=0;             n_fi_next[i]=0;
vaux11[i]=0;          vaux12[i]=0;
vaux30[i]=0;   vaux31[i]=0;   vaux32[i]=0;
vaux33[i]=0;   vaux34[i]=0;   vaux35[i]=0;
vaux36[i]=0;   vaux37[i]=0;   vaux38[i]=0;
vaux39[i]=0;   vaux40[i]=0;   vaux41[i]=0;
vaux42[i]=0;   vaux43[i]=0;   vaux44[i]=0;
vaux45[i]=0;

fcort_p[i]=0;          fcort_n[i]=0;
}

for (i=1;i<num_files+1;i++) {
ni[i]=0;
}

} /* FIM do for de leitura dos n arquivos u.n.inp */

printf("\nterminei de ler arquivos,
vou escrever saidas \n\n");

/* fecha arquivos de escrita */
fclose(write_ni);
fclose(write_delta_press);
fclose(write_flow_out);

free(line2);

printf("\n\n\tFIM !!!\n\n");
system("play shutdown1.wav");
return 0;
}

```

## 7.2 Algoritmo de controle de acionamento da bomba

Esta seção descreve o algoritmo de implementação do controle de acionamento da bomba.

```
/*                                                    */
/*                CONTROLE DE POSICAO                */
/*                                                    */
/*                Marco de 2008                       */
/*                Flavio Jose Aguiar Soares          */
/*                                                    */
/* Comentarios tecnicos:                            */
/* Ambiente de desenvolvimento = Visual Studio      */
/* Software de controle da placa de aquisicao = NIDAQmx vers. 8.01*/
/*                                                    */
/* Linguagem = ANSI C                                */
/* Leitura e escrita de dados emprega funcoes multicanal */
/*                                                    */
/* Descricao do algoritmo:                           */
/* Este programa implementa um Controle Proporcional. */
/*                                                    */
/* O algoritmo deve controlar a posicao de uma viga flexivel, */
/* que e utilizada como o elemento propulsor de uma bomba */
/* dinamica, de modo a produzir uma onda senoide que se propague */
/* com parametros controlados ao longo da viga.      */
/*                                                    */
/* O controle e implementado utilizando uma placa de aquisicao */
/* National Instruments PCI 6229, utilizando Ni-DAQ, C++ ANSI e */
/* utiliza funcoes de controle da placa fornecidos pelo */
/* fabricante.                                        */
/*                                                    */
/* A acao de controle eh implementada acionando-se seis pares de */
/* solenoides atraves das saidas analogicas da placa de aquisicao.*/*
/* Como a placa possui 4 saidas analogicas e temos 12 atuadores, */
/* os 4 sinais analogicos sao demultiplexados para os 12 enderecos*/
/* de hardware. Para isso e empregado uma placa eletronica para a */
/* demultiplexacao e outra para o acionamento das bobinas.      */
/*                                                    */
/* A realimentacao da posicao de cinco pontos da viga e feito com */
/* cinco sensores capacitivos diferenciais que sao lidos atraves */
/* de um circuito que gera um sinal analogico proporcional a */
/* posicao da lamina                                             */
/*                                                    */
/*                                                    */
/* Todas as unidades estão no sistema Internacional - SI      */
/*                                                    */
/*                                                    */
/* Hipotese de Controle:                                     */
```

```

/*****
/* W(x,t)->---Erro----->-Controle--->-Atuadores-----|--> */
/*          ^                                     |      */
/*          |                                     |      */
/*          |                                     |      */
/*          -<-----Sensores-<----- */
*****/

/* Serao implementados os seguintes passos: */
/* */
/* 1 - Geracao do vetor y_ref; */
/* */
/* 2 - Geracao do vetor y_med; */
/* 2.1 - Entrada de Sinal Analogico - Leitura dos sensores; */
/* 2.2 - Converte sinal dos sensores para deslo. nos atuadores; */
/* */
/* 3 - Calculos da malha de controle - Geracao do vetor de forcas; */
/* 3.1 - Conversao corrente -> tensao para drives de acionamento; */
/* */
/* 4 - Saida de Sinal Analogico - acionamento dos atuadores; */
/* 4.1 - Enderecamento do Sinal Digital para a placa demultiplex */
/* */
/* 5 - Realimentacao do modelo em Malha Fechada */
/* */
*****/
*
*****/
#include <sys/time.h>
#include <sys/types.h>
#include <time.h>
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include <NIDAQmx.h>
#include "nrutil.h"
#define pi 3.1415926535897931
#define beta 0.001
#define largura 0.01

/* Variaveis Globais */
float64 sensor = 0; /* retorna tensao no sensor de posicao */

/* Types do software da placa NI 6229 */
TaskHandle taskHandle_ai = 0;
TaskHandle taskHandle_ao = 0;
TaskHandle taskHandle_dio = 0;

/* Funcao que comanda placa Demultiplex - 4 entradas, 12 saidas */
/* Muda endereco digital das saidas analogicas */

```

```

void digital_output(int endereco)
{
uInt8      data_dio[8]={0,0,0,0,0,0,0,0};
/*
if (endereco == 0){
    data_dio[0] = 0; /* pino saida = pino port0_0 */
    data_dio[2] = 0; /* pino saida = pino port0_2 */
    data_dio[3] = 0; /* pino saida = pino port0_4 */
    //printf("endereco = (%d,%d,%d) \n\n",data_dio[0],data_dio[2],
        data_dio[3]);
}
/*
if (endereco == 1){
    data_dio[0] = 1; /* pino saida = pino port0_0 */
    data_dio[2] = 0; /* pino saida = pino port0_2 */
    data_dio[3] = 0; /* pino saida = pino port0_4 */
    //printf("endereco = (%d,%d,%d) \n\n",data_dio[0],data_dio[2],
        data_dio[3]);
}

if (endereco == 2){
    data_dio[0] = 0; /* pino saida = pino port0_0 */
    data_dio[2] = 1; /* pino saida = pino port0_2 */
    data_dio[3] = 0; /* pino saida = pino port0_4 */
    //printf("endereco = (%d,%d,%d) \n\n",data_dio[0],data_dio[2],
        data_dio[3]);
}

    data_dio[4] = 1; /* enable = 1; disable = 0*/

DAQmxWriteDigitalLines(taskHandle_dio,1,0,10.0,
DAQmx_Val_GroupByChannel,data_dio,NULL,NULL);

return;
}

int main(void)
{
    int i=0,j=0,k=0,a=1,m=0;
/* Valores iniciais para calibracao
/* p1 e t1 sao relativos a bobinas impares
/* p2 e t2 sao relativos a bobinas pares
float p1[5]={0.004,0.004,0.004,0.004,0.004};
float p2[5]={-0.004,-0.004,-0.004,-0.004,-0.004};
float t1[5]={-0.20,0.95,1.03,1.07,0.68};
float t2[5]={0.38,-0.23,0.19,2.27,0.25};

float tensao0,tensao1,tensao2,tensao3,tensao4;
tensao0=0.0;
tensao1=0.0;
tensao2=0.0;

```

```

tensao3=0.0;
tensao4=0.0;

float maior0,maior1,maior2,maior3,maior4;
maior0=t2[0]; // assume valores pre-calibrados
maior1=t1[1];
maior2=t1[2];
maior3=t2[3];
maior4=t1[4];

float menor0,menor1,menor2,menor3,menor4;
menor0=t1[0]; // assume valores pre-calibrados
menor1=t2[1];
menor2=t2[2];
menor3=t1[3];
menor4=t2[4];

float maior_que0,maior_que1,maior_que2,maior_que3,maior_que4;
maior_que0=0.0;
maior_que1=0.0;
maior_que2=0.0;
maior_que3=0.0;
maior_que4=0.0;

float menor_que0,menor_que1,menor_que2,menor_que3,menor_que4;
menor_que0=0.0;
menor_que1=0.0;
menor_que2=0.0;
menor_que3=0.0;
menor_que4=0.0;

FILE *fp;
fp=fopen("erro.dat","w");

/* Parametros que serao passados como argumentos de funcoes      */
uInt8 endereco = 0; /* parametro de funcao, indica pinos da placa */

/* Parametros de configuracao da forma de onda na Viga:          */
float amp = 0.0025; /* amplitude */
float ang = 0.05; /* comprimento de onda na lamina [rad] */
float dt = 0.003; /* nyquist da freq. do 3o. modo */
float fase = 5.0; /* vel. angular de propagacao da onda [Hz] */
float cv = 0.20; /* comprimento da lamina [m] */
float mi_b = 1.2566e-6; /* [N/A] */

/* Parametros ajustaveis do Controle da Viga:                    */
int64 tf = 0; /* num. de identif. da interacao temporal */
int64 ciclos = 0; /* val. aux. p/ limitar n. de graus no tempo */
float tempo = 0; /* tempo de acionamento da bomba */
float ne = 2100; /* numero de espiras da bobina do eletroima */
float dbob = 0.015; /* dim.o da secao do nucleo das bobina [m] */

```

```

float gsa = 12.98;    /* ganho de saida do atuador 5V/0.385 mA */
float64 tensao = 0;  /* recebe sinal analog. do sensor de posic. */

/* Declaracao de Matrizes */

/* Matrizes globais de massa [Mf] e matriz de rigidez [Kf] obtidas*/
/* a partir de um modelo MEF do programa FELT. */

/* O FELT foi resolvido considerando que cada noh e composto por */
/* (dx,dy,dw_dx), ou seja, um deslocamento em x, */
/* um deslocamento em y e uma rotacao em z. */

/* A imposicao das condicoes de contorno faz com que a matriz */
/* de massa global e a matriz de rigidez global tenham */
/* suas dimensoes reduzidas de (24X24) -> (21X21) */

/*          dx1 = 0; */
/*          dy1 = 0; */
/*          dy8 = 0; */

float alfa[6][6]={
beta, 0, 0, 0, 0, 0,
0, beta, 0, 0, 0, 0,
0, 0, beta, 0, 0, 0,
0, 0, 0, beta, 0, 0,
0, 0, 0, 0, beta, 0,
0, 0, 0, 0, 0, beta
};/*

/* Declaracao dos vetores */

/* vetor das forcas para os atuadores */
float forca[6]={0,0,0,0,0,0};

/* vetor da corrente eletrica sobre os atuadores eletromaneticos */
double corr[6]={0,0,0,0,0,0};

/* vetor das tensao eletrica sobre os atuadores eletromaneticos */
float bob_aux[6]={0,0,0,0,0,0};

/* impoe tensao em pino de saida analogico */
float tensao_saida[6]={0,0,0,0,0,0};

/* define o sentido da forca sobre a lamina */
int sinal[6] = {1,1,1,1,1,1};

/* vetor de estados no instante (k) */
float y_med[12]={0,0,0,0,0,0,0,0,0,0,0,0};

/* sinal de referencia */
float y_ref[12]={0,0,0,0,0,0,0,0,0,0,0,0};

```

```

/* sinal de referencia no instante anterior */
float y_ref_ant[12]={0,0,0,0,0,0,0,0,0,0,0,0};

/* variacao do sinal de referencia */
float delta_ref[12]={0,0,0,0,0,0,0,0,0,0,0,0};

/* posicao dos atuadores [m] */
float nodes[8] = {0.000,0.021,0.056,0.091,0.126,0.161,0.196,0.200};

/* posicao dos sensores [m] */
float capac[5] = {0.040, 0.075, 0.110, 0.145, 0.180};

/* tensao sobre os sensores */
float posicao_da_lamina[5]={0,0,0,0,0};

/* sinal de tensao lido no ciclo anterior */
float data_anterior[5]={0,0,0,0,0};

/* distancia entre o atuador e a lamina */
float dist[6]={0,0,0,0,0,0};

/* vetor deslocamentos na direcao y, na posicao dos atuadores: */
float wy[8]={0,0,0,0,0,0,0,0};

/* matrizes auxiliares */
float err[12]={0,0,0,0,0,0,0,0,0,0,0,0};

/* Declaracoes para a leitura dos sinais de entrada analogica */
/* Analog Input */
int32 read;
float64 data_ai[5];

/* Declaracoes para a saida dos sinais de escrita analogica */
/* Analog Output */
int32 written;
float64 data_ao[4];

/* ----- */
/* Inicio do algoritmo de controle */
/* ----- */

fase = fase*2*pi; /* Converte unidade fase [Hz] -> [rad/s] */

/* Declaracoes da funcao de Analog_Input */

DAQmxCreateTask("",&taskHandle_ai);
DAQmxCreateAIVoltageChan(taskHandle_ai,"Dev1/ai1,Dev1/ai3,Dev1/ai6,
Dev1/ai9,Dev1/ai11","",DAQmx_Val_RSE,-5.0,5.0,DAQmx_Val_Volts,NULL);
DAQmxCfgSampClkTiming(taskHandle_ai,"",100,DAQmx_Val_Rising,
DAQmx_Val_ContSamps,5);

```

```

DAQmxStartTask(taskHandle_ai);

/* Declaracoes da funcao de Analog_Output */
DAQmxCreateTask("", &taskHandle_ao);
DAQmxCreateAOVoltageChan(taskHandle_ao, "Dev1/ao0:3", "", -5.0, 5.0,
DAQmx_Val_Volts, NULL);
DAQmxCfgSampClkTiming(taskHandle_ao, "", 100, DAQmx_Val_Rising,
DAQmx_Val_FiniteSamps, 5);
DAQmxStartTask(taskHandle_ao);

/* Declaracoes de Digital_Output */
DAQmxCreateTask("", &taskHandle_dio);
DAQmxCreateDOChan(taskHandle_dio, "Dev1/port0/line0:7", "",
DAQmx_Val_ChanForAllLines);
DAQmxStartTask(taskHandle_dio);

/* ----- */
/*                               Inicio de ciclo de controle infinito          */
/* ----- */
m=0;
// for(;;){ // Ciclo Infinito
// m=m+1;
for (m=0;m<=12000;m++){ //Ciclo Finito
// printf("estou imprimindo o ciclo numero %d \n\n",m);
/* ----- */
/* Geracao do vetor y_ref - vetor de estado de referencia          */
/* ----- */

/* Calc. dos desloc. (wy) nodais e vel. (ddt_wy) nodais em y          */
for (i=0;i<=7;i++){
wy[i] = amp*(1-exp(-nodes[i]*100))*
(1-exp((nodes[i]-cv)*100))*sin((pi/ang)*
nodes[i]-fase*tf*dt);
}

/* tempo inicial tf = 0 */
tf = tf + 1; /* incremento da interacao temporal          */
tempo = dt*tf;

/* salva valor da ultima interacao */
y_ref_ant[0] = y_ref[0];          /* no 2 */
y_ref_ant[1] = y_ref[1];          /* no 3 */
y_ref_ant[2] = y_ref[2];          /* no 4 */
y_ref_ant[3] = y_ref[3];          /* no 5 */
y_ref_ant[4] = y_ref[4];          /* no 6 */
y_ref_ant[5] = y_ref[5];          /* no 7 */

y_ref[0] = wy[1];          /* no 2 */
y_ref[1] = wy[2];          /* no 3 */
y_ref[2] = wy[3];          /* no 4 */
y_ref[3] = wy[4];          /* no 5 */

```



```

y_ref[4] = wy[5];          /* no 6 */
y_ref[5] = wy[6];          /* no 7 */

/* ----- */
/* Geracao do vetor y_med - vetor de estado medido      */
/* ----- */

/* Entrada de Sinal Analogico - Leitura dos sensores de posicao. */

/* A funcao de interpolacao deve converter              */
/* o sinal de tensao em deslocamento.                  */

//Leitura dos sinais analog. dos sensores p/ tensao absoluta - RSE
DAQmxReadAnalogF64(taskHandle_ai,-1,0,DAQmx_Val_GroupByChannel,
data_ai,5,&read,NULL);

// Rotina de auto-calibracao,
// procura pelo maior e menor valor de tensao lidos,
// compara com valores de pre-calibracao.
// A saida sao sinais que serao usados para fazer a interpolacao
// de primeira ordem e relacionar a tensao medida nos sensores
// com a posicao da lamina relativa aquela posicao

// salva leitura anterior, para poder fazer as comparacoes
data_anterior[0] = data_ai[0];
data_anterior[1] = data_ai[1];
data_anterior[2] = data_ai[2];
data_anterior[3] = data_ai[3];
data_anterior[4] = data_ai[4];

// auto-calibra s1
if (data_ai[0] >= data_anterior[0]){
    maior_que0 = data_ai[0];
    if (maior_que0 >= maior0){
        maior0 = (maior0 + maior_que0)/2;
        t2[0] = 0.9* maior0;
    }
}
if (data_ai[0] < data_anterior[0]){
    menor_que0 = data_ai[0];
    if (menor_que0 < menor0){
        menor0 = (menor0 + menor_que0)/2;
        t1[0] = 0.9* menor0;
    }
}

// auto-calibra s2
if (data_ai[1] >= data_anterior[1]){
    maior_que1 = data_ai[1];
    if (maior_que1 >= maior1){

```

```
        maior1 = (maior1 + maior_que1)/2;
        t1[1] = 0.9* maior1;
    }
}
if (data_ai[1] < data_anterior[1]){
    menor_que1 = data_ai[1];
    if (menor_que1 < menor1){
        menor1 = (menor1 + menor_que1)/2;
        t2[1] = 0.9* menor1;
    }
}

// auto-calibra s2
if (data_ai[2] >= data_anterior[2]){
    maior_que2 = data_ai[2];
    if (maior_que2 >= maior2){
        maior2 = (maior2 + maior_que2)/2;
        t1[2] = 0.9* maior2;
    }
}
if (data_ai[2] < data_anterior[2]){
    menor_que2 = data_ai[2];
    if (menor_que2 < menor2){
        menor2 = (menor2 + menor_que2)/2;
        t2[2] = 0.9* menor2;
    }
}

// auto-calibra s3
if (data_ai[3] >= data_anterior[3]){
    maior_que3 = data_ai[3];
    if (maior_que3 >= maior3){
        maior3 = (maior3 + maior_que3)/2;
        t2[3] = 0.9* maior3;
    }
}
if (data_ai[3] < data_anterior[3]){
    menor_que3 = data_ai[3];
    if (menor_que3 < menor3){
        menor3 = (menor3 + menor_que3)/2;
        t1[3] = 0.9* menor3;
    }
}

// auto-calibra s4
if (data_ai[4] >= data_anterior[4]){
    maior_que4 = data_ai[4];
    if (maior_que4 >= maior4){
        maior4 = (maior4 + maior_que4)/2;
        t1[4] = 0.9* maior4;
    }
}
```

```

}
if (data_ai[4] < data_anterior[4]){
    menor_que4 = data_ai[4];
    if (menor_que4 < menor4){
        menor4 = (menor4 + menor_que4)/2;
        t2[4] = 0.9* menor4;
    }
}

/* O valor da tensao foi corrigido p/ a posicao central da curva */
/* de interpolacao, ou seja, o centro da curva corresponde a */
/* tensao de 0,5V. A relacao posicao X tensao foi medida de um */
/* lado para o outro do canal, assim, a posicao central */
/* eh a posicao medida deslocada de meio canal */

    tensao = (data_ai[0] + tensao0)/2.0;
    tensao0 = tensao;
    posicao_da_lamina[0] =
        ((p2[0]-p1[0])*(tensao-t1[0]))/(t2[0]-t1[0])+p1[0];
// printf("sensor1 = %.4f posicao=%.4f %ld\n\n",tensao,
    posicao_da_lamina[0],m);

    tensao = (data_ai[1] + tensao1)/2.0;
    tensao1=tensao;
    posicao_da_lamina[1] =
        ((p2[1]-p1[1])*(tensao-t1[1]))/(t2[1]-t1[1])+p1[1];
// printf("sensor2 = %.4f posicao=%.4f %ld\n\n",tensao,
    posicao_da_lamina[1],m);

    tensao = (data_ai[2] + tensao2)/2.0;
    tensao2=tensao;
    posicao_da_lamina[2] =
        ((p2[2]-p1[2])*(tensao-t1[2]))/(t2[2]-t1[2])+p1[2];
// printf("sensor3 = %.4f posicao=%.4f %ld\n\n",tensao,
    posicao_da_lamina[2],m);

    tensao = (data_ai[3] + tensao3)/2.0;
    tensao3=tensao;
    posicao_da_lamina[3] =
        ((p2[3]-p1[3])*(tensao-t1[3]))/(t2[3]-t1[3])+p1[3];
// printf("sensor4 = %.4f posicao=%.4f %ld\n\n",tensao,
    posicao_da_lamina[3],m);

    tensao = (data_ai[4] + tensao4)/2.0;
    tensao4=tensao;
    posicao_da_lamina[4] =
        ((p2[4]-p1[4])*(tensao-t1[4]))/(t2[4]-t1[4])+p1[4];
// printf("sensor5 = %.4f posicao=%.4f %ld\n\n",tensao,
    posicao_da_lamina[4],m);

/* A tensao medida deve ser usada para a calibracao */

```

```

    for (i=0;i<5;i++){
//   printf("tensao[i]=%f, i=%d \n\n",data_ai[i],i);
    }

/* Converte sinais de tensao dos 5 sensores p/ desloc. nos nohs; */
/* Conhecendo os deslocamentos nos 6 nohs em frente aos atuadores,*/
/* prossegue e realimenta o algoritmo.                                     */

// posicao dos nohs
// float nodes[8]={0.000,0.021,0.056,0.091,0.126,0.161,0.196,0.200};
// posicao dos sensores [m]
// float capac[5] = {0.040, 0.075, 0.110, 0.145, 0.180};

y_med[0]=0.20*posicao_da_lamina[0];
y_med[1]=((posicao_da_lamina[3]-posicao_da_lamina[4])/0.035)*0.016
        + posicao_da_lamina[4];

y_med[2]=((posicao_da_lamina[2]-posicao_da_lamina[3])/0.035)*0.016
        + posicao_da_lamina[3];

y_med[3]=((posicao_da_lamina[1]-posicao_da_lamina[2])/0.035)*0.016
        + posicao_da_lamina[2];

y_med[4]=((posicao_da_lamina[0]-posicao_da_lamina[1])/0.035)*0.016
        + posicao_da_lamina[1];

y_med[5]=0.5250*posicao_da_lamina[0];

/* ----- */
/* Geracao das forcas sobre os atuadores - aplicadas sobre os nos*/
/* ----- */

/* err[i] = y_ref - y_med; err = erro de posicao da lamina nos */

err[0] = y_ref[0] - y_med[0]; /* erro de posicao no noh 1 */
err[1] = y_ref[1] - y_med[1]; /* erro de posicao no noh 2 */
err[2] = y_ref[2] - y_med[2]; /* erro de posicao no noh 3 */
err[3] = y_ref[3] - y_med[3]; /* erro de posicao no noh 4 */
err[4] = y_ref[4] - y_med[4]; /* erro de posicao no noh 5 */
err[5] = y_ref[5] - y_med[5]; /* erro de posicao no noh 6 */
err[6] = y_ref[6] - y_med[6]; /* erro de velocidade no noh 1 */
err[7] = y_ref[7] - y_med[7]; /* erro de velocidade no noh 2 */
err[8] = y_ref[8] - y_med[8]; /* erro de velocidade no noh 3 */
err[9] = y_ref[9] - y_med[9]; /* erro de velocidade no noh 4 */
err[10] = y_ref[10] - y_med[10]; /* erro de velocidade no noh 5 */
err[11] = y_ref[11] - y_med[11]; /* erro de velocidade no noh 6 */

forca[0] = (alfa[0][0] * err[0]);
forca[1] = (alfa[1][1] * err[1]);
forca[2] = (alfa[2][2] * err[2]);
forca[3] = (alfa[3][3] * err[3]);

```

```

forca[4] = (alfa[4][4] * err[4]);
forca[5] = (alfa[5][5] * err[5]);

/* ----- */
/* Saida do Sinal Analog. - Acionamento dos atuadores eletromags. */
/* ----- */

/* Calculo da corrente p/ geracao da forca magnetica nas bobinas */
for (k=0;k<=5;k++){
// printf("y_ref[k] = %f, y_med =%f, ki = %d \n\n",y_ref[k],
y_med[k],k);
    sinal[k]=1.0;
    if (forca[k] < 0){
forca[k] = fabs(forca[k]);
    }

    if (y_ref[k] > 0){
        dist[k] = largura/2 - y_med[k];
    }
    if (y_ref[k] < 0)
        { dist[k] = largura/2 + y_med[k];
    }

    delta_ref[k] = y_ref[k] - y_ref_ant[k];

    if (delta_ref[k] > 0) {
sinal[k] = -1.0;
    }

fprintf(fp,"%f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f
        %f %d %d %d %d %d %d\n",
        y_ref[0],y_ref[1],y_ref[2],y_ref[3],y_ref[4],y_ref[5],
        y_med[0],y_med[1],y_med[2],y_med[3],y_med[4],y_med[5],
        posicao_da_lamina[0],
        posicao_da_lamina[1],
        posicao_da_lamina[2],
        posicao_da_lamina[3],
        posicao_da_lamina[4],
        sinal[0],
        sinal[1],
        sinal[2],
        sinal[3],
        sinal[4],
        sinal[5]);

    corr[k] = ((2*(dist[k])) / (mi_b*ne*dbob))*(sqrt(forca[k]));
    printf("corr[k] = %f, ki = %d \n\n",corr[k],k);

if ((corr[k] > 0.400) || (corr[k] < -0.400)){
    corr[k] = 0.400;
}

```

```

//      printf("corr[k] = %f, ki = %d \n\n",corr[k],k);

      /* Conversao corrente tensao para drives de acionamento      */
      /* Converte [A]->[V]                                          */
      tensao_saida[k] = corr[k]*gsa;

      /* Rotina para proteger a placa de aquisicao                  */
      if (tensao_saida[k] > 5.00){
      tensao_saida[k] = 5.00;
      }

      if (tensao_saida[k] < -5.00){
      tensao_saida[k] = 5.00;
      }

      if (tensao_saida[k] < -0.00){
      tensao_saida[k] = fabs(tensao_saida[k]);
      }

//printf("cor[k]=%f er[k]=%f for[k]=%f tensao[k]=%f noh=%d \n",
corr[k],err[k],forca[k]*sinal[k],tensao_saida[k],k);

    } // Fim do for das forcas

/* ----- */
/* Enderecamento do Sinal Digital para a placa de demultiplex.  */
/* Cada end. transfere 4 sinais analog. p/ a placa demultiplex.  */

/* Escrita do sinal analogico nos pinos de saida analog. da placa */

/* Habilita endereco 0 = {0,0,0}                                  */
endereco = 0; /* Aciona bobinas 0, 1, 2 e 3 */
digital_output(endereco);
if (sinal[5] > 0){ /* bobinas 0 e 1 */
    data_ao[0] = tensao_saida[0];
    data_ao[1] = 0;
// printf("tensao_saida[0]=%f na bobina %d \n\n",tensao_saida[0],0);
}
if (sinal[5] < 0){
    data_ao[0] = 0;
    data_ao[1] = tensao_saida[0];
// printf("tensao_saida[0]=%f na bobina %d \n\n",tensao_saida[0],1);
}

if (sinal[4] > 0){ /* bobinas 2 e 3 */
    data_ao[2] = tensao_saida[1];
    data_ao[3] = 0;
// printf("tensao_saida[1]=%f na bobina %d \n\n",tensao_saida[1],2);
}
if (sinal[4] < 0){
    data_ao[2] = 0;

```

```

        data_ao[3] = tensao_saida[1];
//    printf("tensao_saida[1]=%f na bob %d \n\n",tensao_saida[1],3);
    }
    for(k=0;k<=3;k++){
//    printf("data_ao[k]=%f, no canal %d \n\n",data_ao[k],k);
    }

/* Escreve o vetor dos sinais analogicos data_ao[i] no endereco 0 */
DAQmxWriteAnalogF64(taskHandle_ao,1,0,0,DAQmx_Val_GroupByChannel,
data_ao,&written,NULL);

/* Habilita endereco 0 = {1,0,0} */
    endereco = 1;    /* Aciona bobinas 4, 5, 6 e 7 */
    digital_output(endereco);
    if (sinal[3] > 0){    /* bobinas 4 e 5 */
        data_ao[0] = tensao_saida[2];
        data_ao[1] = 0;
//    printf("tensao_saida[2]=%f na bob %d \n\n",tensao_saida[2],4);
    }
    if (sinal[3] < 0){
        data_ao[0] = 0;
        data_ao[1] = tensao_saida[2];
//    printf("tensao_saida[2]=%f na bob %d \n\n",tensao_saida[2],5);
    }

    if (sinal[2] > 0){    /* bobinas 6 e 7 */
        data_ao[2] = tensao_saida[3];
        data_ao[3] = 0;
//    printf("tensao_saida[3]=%f na bob %d \n\n",tensao_saida[3],6);
    }
    if (sinal[2] < 0){
        data_ao[2] = 0;
        data_ao[3] = tensao_saida[3];
//    printf("tensao_saida[3]=%f na bob %d \n\n",tensao_saida[3],7);
    }
    for(k=0;k<=3;k++){
//    printf("data_ao[k]=%f, no canal %d \n\n",data_ao[k],k);
    }

/* Escreve o vetor dos sinais analogicos data_ao[i] no endereco 1 */
DAQmxWriteAnalogF64(taskHandle_ao,1,0,0,DAQmx_Val_GroupByChannel,
data_ao,&written,NULL);

/* Habilita endereco 0 = {0,1,0} */
    endereco = 2;    /* aciona bobinas 8, 9, 10 e 11 */
    digital_output(endereco);
    if (sinal[1] > 0){    /* bobinas 8 e 9 */
        data_ao[0] = tensao_saida[4];
        data_ao[1] = 0;

```

```

//   printf("tensao_saida[4]=%f na bob %d \n\n",tensao_saida[4],8);
//   }
//   if (sinal[1] < 0){
//       data_ao[0] = 0;
//       data_ao[1] = tensao_saida[4];
//   printf("tensao_saida[4]=%f na bob %d \n\n",tensao_saida[4],9);
//   }

//   if (sinal[0] > 0){      /* bobinas 10 e 11 */
//       data_ao[2] = tensao_saida[5];
//       data_ao[3] = 0;
//   printf("tensao_saida[5]=%f na bob %d \n\n",tensao_saida[5],10);
//   }
//   if (sinal[0] < 0){
//       data_ao[2] = 0;
//       data_ao[3] = tensao_saida[5];
//   printf("tensao_saida[5]=%f na bob %d \n\n",tensao_saida[5],11);
//   }
//   for(k=0;k<=3;k++){
//   printf("data_ao[k]=%f, no canal %d \n\n",data_ao[k],k);
//   }

/* Escreve o vetor dos sinais analogicos data_ao[i] no endereco 2 */
DAQmxWriteAnalogF64(taskHandle_ao,1,0,0,DAQmx_Val_GroupByChannel,
data_ao,&written,NULL);

    sinal[0] = 1; /* restabelece sinais para reiniciar ciclo */
    sinal[1] = 1;
    sinal[2] = 1;
    sinal[3] = 1;
    sinal[4] = 1;
    sinal[5] = 1;

    ciclos = ciclos + 1;
    //printf("\n\n\n tempo = %f [s], ciclos=%d, \n\n",tempo,ciclos);

/*****
/* Contagem do tempo do ciclo de */

usleep(1975);//dt=0.005 => usllep=1975 + for = 50000
j=0;
for (i=1;i<50000;i++) {
    j=j+1;
}

}          /*          Fim do ciclo infinito          */

printf("vou fechar os tasks \n\n");

if( taskHandle_ai !=0 ) {
DAQmxStopTask(taskHandle_ai);

```



```
DAQmxClearTask(taskHandle_ai);
}

if( taskHandle_ao !=0 ) {
    DAQmxStopTask(taskHandle_ao);
    DAQmxClearTask(taskHandle_ao);
}

if( taskHandle_dio !=0 ) {
    DAQmxStopTask(taskHandle_dio);
    DAQmxClearTask(taskHandle_dio);
}

printf("Agora devo reiniciar o ciclo \n\n");
fclose(fp);
return;
}
```

## 8 APÊNDICE C - Algoritmos em FORTRAN77

### 8.1 KL1.0

Este arquivo define a malha de elementos finitos inicial. Possui um parágrafo de coordenadas nodais, seguido de um parágrafo de topologia e termina com um parágrafo de identificação da fronteira a que pertence determinado nó.

```
Grobgitter KL1.0 - Kanal mit 0 Loch
Parametrisierung PARQL
      24 39 0 4 1          NEL NVT NMT NVE NBCT
DCORVG
      0D0      0D0
49.5D0      0D0
      49D0      0D0
      2D0      0D0
      2D0      0.5D0
      47D0      0D0
      4D0      0D0
      4D0      0.5D0
      45D0      0D0
      6D0      0D0
      6D0      0.5D0
      43D0      0D0
      8D0      0D0
      8D0      0.5D0
      41D0      0D0
      10D0      0D0
      10D0      0.5D0
      39D0      0D0
      12D0      0D0
      12D0      0.5D0
      37D0      0D0
      14D0      0D0
      14D0      0.5D0
      35D0      0D0
      16D0      0D0
```

16D0	0.5D0
33D0	0D0
18D0	0D0
18D0	0.5D0
31D0	0D0
20D0	0D0
20D0	0.5D0
29D0	0D0
22D0	0D0
22D0	0.5D0
27D0	0D0
24D0	0D0
24.5D0	0D0
25D0	0D0

## KVERT

1	4	5	2
2	5	6	3
4	7	8	5
5	8	9	6
7	10	11	8
8	11	12	9
10	13	14	11
11	14	15	12
13	16	17	14
14	17	18	15
16	19	20	17
17	20	21	18
19	22	23	20
20	23	24	21
22	25	26	23
23	26	27	24
25	28	29	26
26	29	30	27
28	31	32	29
29	32	33	30
31	34	35	32
32	35	36	33
34	37	38	35
35	38	39	36

## KNPR

1	1	1	1	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1	
0	1	1	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1	1	1

## KMM

1	2
---	---

## 8.2 *parq2d.f*

Arquivo de definições do contorno da fronteira externa da malha inicial de elementos finitos.

```
      DOUBLE PRECISION FUNCTION PARX(T, IBCT)
      IMPLICIT REAL*8 (A-H, O-Z)
C
      IF (T.GE.49.0D0) GOTO 3
      IF (T.GE.25.0D0) GOTO 2
      IF (T.GE.24.0D0) GOTO 1
      PARX=T
      GOTO 99999
1     PARX=24.0D0
      GOTO 99999
2     PARX=49.0D0-T
      GOTO 99999
3     PARX=0.0D0
      GOTO 99999
C
99999 END
C
      DOUBLE PRECISION FUNCTION PARY(T, IBCT)
      IMPLICIT REAL*8 (A-H, O-Z)
C
      IF (T.GE.49.0D0) GOTO 3
      IF (T.GE.25.0D0) GOTO 2
      IF (T.GE.24.0D0) GOTO 1
      PARY=0.0D0
      GOTO 99999
1     PARY=T-24.0D0
      GOTO 99999
2     PARY=1.0D0
      GOTO 99999
3     PARY=50.0D0-T
      GOTO 99999
C
99999 END
C
      DOUBLE PRECISION FUNCTION TMAX(IBCT)
      IMPLICIT REAL*8 (A-H, O-Z)
      TMAX=50.0D0
      END
```

## 8.3 indat2d.f

Arquivo de definições das condições de contorno em torno do volume de controle.

```

*****
DOUBLE PRECISION FUNCTION FDATIN(ITYP,IBLOC,X,Y,TIMENS,RE,TSTEP)
*
*   Prescribed data for files coeff.f and bndry.f
*****
      IMPLICIT DOUBLE PRECISION(A,C-H,O-U,W-Z),LOGICAL(B)
      PARAMETER (PI=3.1415926535897931D0)

C
C
C
      FDATIN=ODO

C
C
C
C=====
C *** Case 1: Velocity boundary values and/or exact solution
C=====
C
C   DRAD define o comprimento da lâmina
C
C   DEPS define a espessura da lâmina, deve ser < 1.5
C   a lâmina fica no centro
C
C   DVEL define a velocidade angular
C
C   DANG define a amplitude de oscilação da lâmina
C   DANG = 0.25 corresponde a 90 graus
C
C   DFASE = velocidade de propagacao longitudinal da onda
C
C   XM1 e YM1 são as coordenadas do ponto
C   de rotação da lâmina
C
C   se FDATIN=ODO, então programa trava, solução diverge.
C   para resolver NS é preciso introduzir uma
C   velocidade inicial
C
C   PPP é a distância entre a origem (XM1,YM1)
C   e a coordenada do nó (XX,YY)
C
C   PHI1 é uma senóide que controla o movimento
C   oscilatório da lâmina, limitada por DANG
C
C   PHI5 é o angulo formado ente a origem (XM1,YM1)
C   e a coordenada do nó em questão (XX,YY)
C   PHI5 == |

```

```

C
C   FDATIN definida pela equação abaixo, resulta numa
C   constante, cujo módulo depende de PHI5
C
C   FDATIN=-PPP*SIN(PHI5)*DANG*PI*DVEL*COS((TIMENS+1D0)*DVEL)
C
C   PHI1 = função de forma da lâmina
C   todas as unidas no sistema SI
c
  IF (ITYP.equação1) THEN
    DRAD=20D0
    DEPS=0.02D0
    DAMP=0.20D0
    DFASE=18.0D0*2.0D0*PI
    DANG=5.0D0
C
  XM1=2.666D0
  YM1=0.5D0
c
C   VELOCITY COMPONENT VX
  IF (IBLOC.equação1) THEN
    PPP=SQRT(((XX-XM1)**2)+((YY-YM1)**2))
    IF ((PPP.LE.DRAD).AND.(XX.GE.XM1)) THEN
      PHI2=ASIN((YY-YM1)/PPP)
      PI3=DAMP*(1-EXP(-(XX-XM1)))*(1-EXP((XX-XM1)-20))
      PI4=PI3*SIN((PI/DANG)*(XX-XM1)-DFASE*TIMENS)
c      PI4=DAMP*(1-EXP(-(XX-XM1)))*SIN((PI/DANG) ->
c      *(XX-XM1)-DFASE*TIMENS)
      PHI2D=ASIN((PI4)/PPP)
      PHI1D=PHI2-PHI2D
      DIST2=PPP*SIN(PHI1D)
      IF (ABS(DIST2).LE.DEPS) THEN
        FDATIN=0D0
        WRITE(6,*)XX,YY
      ENDIF
    ENDIF
c
  IF (X.equação0.0D0) FDATIN=117.25D0
  IF (Y.LE.0.0D0) FDATIN=0D0
  IF (Y.GE.1.0D0) FDATIN=0D0
  ENDIF
c
C   VELOCITY COMPONENT VY
  IF (IBLOC.equação2) THEN
    PPP=SQRT(((XX-XM1)**2)+((YY-YM1)**2))
    IF ((PPP.LE.DRAD).AND.(XX.GE.XM1)) THEN
      PHI2=ASIN((YY-YM1)/PPP)
      PI3=DAMP*(1-EXP(-(XX-XM1)))*(1-EXP((XX-XM1)-20))
      PI4=PI3*SIN((PI/DANG)*(XX-XM1)-DFASE*TIMENS)
c      PI4=DAMP*(1-EXP(-(XX-XM1)))*SIN((PI/DANG) ->
c      *(XX-XM1)-DFASE*TIMENS)

```

```

      PHI2D=ASIN((PI4)/PPP)
      PHI1D=PHI2-PHI2D
      DIST1=PPP*SIN(PHI1D)
      IF (ABS(DIST1).LE.DEPS) THEN
      PI7=-DFASE*(1-EXP(-(XX-XM1)))*(1-EXP((XX-XM1)-20))
      FDATIN=DAMP*PI7*COS((PI/DANG)*(XX-XM1)-DFASE*TIMENS)
      ENDIF
    ENDIF
  C
      IF (Y.LE.0.0D0) FDATIN=0D0
      IF (Y.GE.1.0D0) FDATIN=0D0
    ENDIF
  C
    ENDIF
  C
  C=====
  C *** Case 2: Velocity x-derivative of exact solution
  C=====
  C
      IF (ITYP.equação2) THEN
  C
          IF (IBLOC.equação1) THEN
              IF (X.equação0.0D0) FDATIN=0D0
          ENDIF
  C
          IF (IBLOC.equação2) THEN
              IF (X.equação0.0D0) FDATIN=0D0
          ENDIF
  C
      ENDIF
  C
  C=====
  C *** Case 3: Velocity y-derivative of exact solution
  C=====
  C
      IF (ITYP.equação3) THEN
  C
          IF (IBLOC.equação1) THEN
              FDATIN=0D0
          ENDIF
  C
          IF (IBLOC.equação2) THEN
              FDATIN=0D0
          ENDIF
  C
      ENDIF
  C
  C=====
  C *** Case 4: Exact pressure solution
  C=====

```

```

C
C     IF (ITYP.equação4) THEN
C
C         FDATIN=ODO
C
C     ENDIF
C
C=====
C *** Case 5: Right hand side for momentum equation
C=====
C
C     IF (ITYP.equação5) THEN
C
C         IF (IBLOC.equação1) THEN
C             FDATIN=ODO
C         ENDIF
C
C         IF (IBLOC.equação2) THEN
C             FDATIN=ODO
C         ENDIF
C
C     ENDIF
C
C=====
C *** Case 6: Right hand side for continuity equation
C=====
C
C     IF (ITYP.equação6) THEN
C
C         FDATIN=ODO
C
C     ENDIF
C
C=====
C *** Case 7: Mean pressure values
C=====
C
C     IF (ITYP.equação7) THEN
C         WRITE(*,*) 'passou pelo ityp 7'
C         DPAR=X
C         INPR=IBLOC
C
C         IF((DPAR.GT.-1.ODO).AND.(DPAR.LT.1.0D-4).AND. ->
C             (INPR.equação1)) THEN
C         IF((DPAR.GT.1D0).AND.(DPAR.LT.2D0).AND. ->
C             (INPR.equação1)) THEN
C             FDATIN=ODO
C         ENDIF
C
C         IF ((DPAR.GT.3D0).AND.(DPAR.LT.4D0).AND. ->
C             (INPR.equação1)) THEN

```



```

          FDATIN=ODO
        ENDIF
C
        ENDIF
C
99999 END
C
*****
      SUBROUTINE NEUDAT(INPART,INPRN,DPARN1,DPARN2,TIMENS)
*
*   Neumann-boundary part
*****
      IMPLICIT DOUBLE PRECISION(A,C-H,O-U,W-Z),LOGICAL(B)
C
C
C
C=====
C *** Case 0: Set number of Neumann-boundary parts
C=====
C
      IF (INPART.equação0) THEN
        INPART=1
      ENDIF
C
C identifica o numero de saidas do volume de controle
C=====
C *** Case <>0: Specify Neumann-boundary parts
C=====
C
      IF (INPART.GT.0) THEN
C
        IF (INPART.equação1) THEN
          DPARN1=24.ODO
          DPARN2=25.ODO
          INPRN =1
        ENDIF
      ENDIF
C
C identifica o intervalo do comprimento normalizado
C   da fronteira
C que corresponde as secoes de saida
99999 END
C
*****
      SUBROUTINE PTSDAT(TIMENS,DNU)
*
*   Data for Point-output (for fpost and bdpres and bdforc)
*****
      IMPLICIT DOUBLE PRECISION(A,C-H,O-U,W-Z),LOGICAL(B)
      COMMON /NSPTS/ KPU(2),KPP(4),KPX(4),KPI(2),DPI(2,2),DPF(2)
      SAVE

```

```

C
      EXTERNAL UE
C
C=====
C *** Points for velocity, pressure and flux-tracing
C=====
C
CCC BENCH4: 24,25,21,22,24,25,11,27
      KPU(1)=53
      KPU(2)=36
C
      KPP(1)=47
      KPP(2)=49
      KPP(3)=53
      KPP(4)=36
C
      KPX(1)=7
      KPX(2)=23
C
C=====
C *** Parameters for 2 integral pressures in bdpres.f
C=====
C
      KPI(1) =1
      DPI(1,1)=0D0
      DPI(2,1)=1D0
C
      KPI(2) =1
      DPI(1,2)=0.00D0
      DPI(2,2)=0.25D0
C
C=====
C *** Parameters for lift (DFW) and drag (DAW) in bdforc.f (INPR=2)
C ***
C *** dfw=2 int_s [dpf(1) dut/dn n_y - p n_x] ds / dpf(2)
C *** daw=2 int_s [dpf(1) dut/dn n_x + p n_y] ds / dpf(2)
C ***
C=====
C
      RHO =0.001D0
      DIST =0.1D0
      UMEAN=1.0D0
C
      DPF(1)=0d0
      DPF(2)=0d0
c      DPF(1)=RHO*DNU
c      DPF(2)=RHO*DIST*UMEAN**2
C
C
C
99999 END

```

## 8.4 **bndry.f**

Arquivo de definição da fronteira móvel e dos nós associados a ela. Segue apenas a parcela do programa que foi alterada para identificar o movimento da lâmina.

```

c      DRAD define o comprimento da lâmina
C
c      DEPS define a espessura da lâmina, deve ser < 1.5
c      a lâmina fica no centro
C
c      DVEL define a velocidade angular
C
c      DANG define a amplitude de oscilação da lâmina
C      DANG = 0.25 corresponde a 90 graus
C
C      DFASE = velocidade de propagacao longitudinal da onda
C
c      XM1 e YM1 são as coordenadas do ponto de rotação da lâmina
C
c      se FDATIN=0D0, então programa trava, solução diverge.
C      para resolver NS é preciso introduzir uma velocidade inicial
c
C      PPP é a distância entre a origem (XM1,YM1)
C      e a coordenada do nó (XX,YY)
C
C      PHI1 é uma senóide que controla o movimento
C      oscilatório da lâmina, limitada por DANG
C
C      PHI5 é o angulo formado ente a origem (XM1,YM1)
C      e a coordenada do nó em questão (XX,YY)
C      PHI5 == |
C
C      FDATIN definida pela equação abaixo, resulta numa
C      constante, cujo módulo depende de PHI5
C
C      FDATIN=-PPP*SIN(PHI5)*DANG*PI*DVEL*COS((TIMENS+1D0)*DVEL)
C
C      PHI1 = função de forma da lâmina
C
C      DCORVRG representa as coordenadas cartesianas dos nós
C
C      KVERT representa a topologia em uma
C      sequência de nós conectados
C
C      KNPR indica a que fronteira pertence o nó,
C
C      KNPR=-2 indica que para valores de espessura
C      menores que DEPS, a fronteira é fictícia.
C
C      KNPR=11 indica os nohs que serao visualizados

```

```

C
C   PHI1 = função de forma da lâmina
C   todas as unidades no sistema SI
c
   DRAD=20D0
   DEPS=0.02D0
   DAMP=0.20D0
   DFASE=18.0D0*2.0D0*PI
   DANG=5.0D0
C
   XM1=2.666D0
   YM1=0.5D0
C
C
DO 2 IEL=1,NEL
DO 20 IVE=1,4
IF (KADJ(IVE, IEL).LT. IEL) THEN
IVT1=KVERT(IVE, IEL)
XX=DCORVG(1, IVT1)
YY=DCORVG(2, IVT1)
C
   PPP=SQRT(((XX-XM1)**2)+((YY-YM1)**2))
   IF ((PPP.LE.DRAD).AND.(XX.GE.XM1)) THEN
       PHI2=ASIN((YY-YM1)/PPP)
       PI3=DAMP*(1-EXP(-(XX-XM1)))*(1-EXP((XX-XM1)-20))
       PI4=PI3*SIN((PI/DANG)*(XX-XM1)-DFASE*TIMENS)
c       PI4=DAMP*(1-EXP(-(XX-XM1)))*SIN((PI/DANG) ->
c       *(XX-XM1)-DFASE*TIMENS)
       PHI2D=ASIN((PI4)/PPP)
       PHI1D=PHI2-PHI2D
       DIST1=PPP*SIN(PHI1D)
       IF (ABS(DIST1).LE.DEPS) THEN
           KNPR(IVT1)=11
       ENDIF
   ENDIF
C
   IVE1=IVE+1
   IF(IVE1.equação5) IVE1=1
   IVT2=KVERT(IVE1, IEL)
   XX1=DCORVG(1, IVT2)
   YY1=DCORVG(2, IVT2)
c
   XX2=0.5D0*(XX+XX1)
   YY2=0.5D0*(YY+YY1)
c
   PPP=SQRT(((XX2-XM1)**2)+((YY2-YM1)**2))
   IF ((PPP.LE.DRAD).AND.(XX2.GE.XM1)) THEN
       PHI2=ASIN((YY2-YM1)/PPP)
       PI3=DAMP*(1-EXP(-(XX2-XM1)))*(1-EXP((XX2-XM1)-20))
       PI4=PI3*SIN((PI/DANG)*(XX2-XM1)-DFASE*TIMENS)
c       PI4=DAMP*(1-EXP(-(XX-XM1)))*SIN((PI/DANG) ->

```

```
C      *(XX-XM1)-DFASE*TIMENS)
      PHI2D=ASIN((PI4)/PPP)
      PHI1D=PHI2-PHI2D
      DIST2=PPP*SIN(PHI1D)
      IF (ABS(DIST2).LE.DEPS) THEN
        IM1=KMID(IVE, IEL)
        KNPR(IM1)=-2
        NMBD=NMBD+1
        KMBD(NMBD)=IM1-NVT
      ENDIF
C
      ENDIF
      ENDIF
C
20    CONTINUE
2    CONTINUE
C
      END
C
```

## 8.5 cc2d.dat

Arquivo de definições e parametrizações dos métodos numéricos. A cada interação o algoritmo de atualização modifica a variável TIMENS deste arquivo, produzindo o avanço temporal da solução do Featflow1.3RC.

```

=====
File for input data for cc2d
=====
unit numbers and file names on /FILES/:
-----
0          IMESH  (0=FEAT-parametrization,1=OMEGA)
0          IRMESH (0=create mesh,>0=read (>1: formatted))
'#pre/bench5.prm' CPARM  (name of parametrization file)
'#adc/KL1.0'    CMESH  (name of coarse mesh file)
'#data/cc2d.out' CFILE  (name of protocol file)
0          ISTART (0=ZERO,1=NLMAX,2=NLMAX-1,- =formatted)
'#data/#DX5'    CSTART (name of start solution file)
0          ISOL   (0=no,1=unformatted,-1=formatted)
'#data/#DX6'    CSOL   (name of final solution file)
-----
Values for /OUTPUT/ :
-----
0  M
0  MT
0  ICHECK
2  MSHOW      (1=file output,2=1+term. output)
-----
Values for /IPARM/,etc :
-----
2  NLMIN      (minimum mg-level)
6  NLMAX      (maximum mg-level)
3  IELT       (0=E031,1=E030,2=E031M,3=E030M)
0  ISTOK      (1=Stokes calculation)
0  IRHS       (0=zero,1=steady inhomog.,2=nonsteady force)
2  IBDR       (0=DIRICHLET,1=Pressure drop,2=NEUMANN part)
0  IERANA     (>0=error analysis (=quadrature formula))
0  IMASS      (0=lumped mass,1=real mass matrix)
0  IMASSL     (0=usual lumping,1=diagonal lumping)
1  IUPW       (1=upwind,0=SD)
0  IPRECA     (0=s-prec,1=d-prec,2/3 from disk for M/ST,4=all new)
3  IPRECB     (0=single,1=double,2=1 on elements for B,3/4=with 2)
3  ICUBM      (cubature formula for M)
4  ICUBA      (cubature formula for ST)
4  ICUBN      (cubature formula for N)
8  ICUBB      ( - " - for B)
1  ICUBF      ( - " - for F)
1  INLMIN     (min nonlinear it.,1=lin extrap.,-1=const extrap.)

```

```

4  INLMAX   (max nonlinear it.,1=lin extrapol.,-1=const extrapol.)
0  ICYCLE   (type of mg-cycle for U: 0=F,1=V,2=W)
1  ILMIN    (minimum of linear mg steps)
5  ILMAX    (maximum of linear mg steps)
2  IINT     (press. interp.:1/-1 const.,2/-2 trilin.)
1  ISM      (U-smoother:1=VANCA)
1  ISL      (U-solver :1=VANCA)
16 NSM      (number of pre/postsmoothing steps)
100 NSL     (maximum of solver-iterations)
2  NSMFAC   (factor for pre/postsm. steps on coarser levels)

```

-----  
Values for /RPARAM/,etc. :

```

-----
100D0 RE      (Viscosity parameter 1/NU)
1.0D0 UPSAM   (parameter for upwind/SD)
1.0D0 OMGMIN  (lower limit for opt. OMEGA,<0 no rel. changes)
1.0D0 OMGMAX  (upper limit for opt. OMEGA,<0 no rel. changes)
1.0D0 OMGINI  (start value limit for opt. OMEGA)
1.0D-5 EPSD   (limit for U-defects)
1.0D-5 EPSDIV (limit for DIV-defects)
1.0D-2 EPSUR  (limit for U-changes)
1.0D-0 EPSPR  (limit for P-changes)
1.0D-1 DMPD   (limit for defect improvement)
5.0D-1 DMPMG  (damping of residuals for mg-it.)
1.0D-1 EPSMG  (limit for residuals for mg-it.)
1.0D-2 DMPSL  (damping of residuals for solving)
1.0D-1 EPSSL  (limit for residuals for solving)
1.0D0 RLXSM   (relaxation for the smoother)
0.8D0 RLXSL   (relaxation for the solver)
1.0D0 AMINMG  (lower limit for optimal MG-ALPHA)
1.0D0 AMAXMG  (upper limit for optimal MG-ALPHA)

```

-----  
Values for /NS.../ :

```

-----
1          ISTAT  (0=steady,1=nonsteady)
1          NITNS  (number of macro time steps)
1D-5      EPSNS  (lower limit for time derivative)
1D-2 TIMENS (absolute start time)
1.0D0     THETA  (parameter for one step scheme)
0.0001D0  TSTEP  (start time step size)
1          IFRSTP (0=one step scheme,1=fractional step)
0          INSAV  (0=no,>0=stepsize for saving sol. vector)
0          INSAVN (modulo number of files (max=10))
0.00000000D0 DTFILM (time difference for film output)
0.00000000D0 DTAVS  (time difference for AVS output)
0.00000000D0 DTGMV  (time difference for BYU output)
0          IFUSAV (level for unformatted velocity)
0          IFPSAV (level for unformatted pressure)
0          IFXSAV (level for unformatted streamlines)
5          IAVS   (level for AVS-output)
5          IGMV   (level for BYU-output)

```

---

```

0          IFINIT  (start file number for film)
-1         IADTIM  (0=no,1=pred(no BNL),2=pred,3=2+rep,<0=+extr)
2.00000D0 TIMEMX  (max. absolute time)
0.00001D0 DTMIN   (min. time step)
0.0001D0  DTMAX   (max. time step)
0.0001D0  DTFACT  (max. changes for time step)
0.0001D0  TIMEIN  (time for start procedure)
5D-3      EPSADI  (low accuracy for acceptance in start)
1D-3      EPSADL  (low accuracy for acceptance after start)
0.50D0    EPSADU  (upper limit for acceptance for |IADTIM|=3)
1         IEPSAD  (1=u(L2),2=u(MX),3=p(L2),4=p(MX),5-8=mix)
2         IADIN   (0=constant,1=linear,2=logarith. start)
1         IREPIT  (max. number of repetitions for |IADTIM|=3)

```

### 8.5.1 Obsevações sobre o uso do Featflow

Ao realizar simulações numéricas com o Featflow1.3RC3 verificou-se as soluções apresentavam maiores resíduos à medida que o tempo de integração avançava. Para contornar esta dificuldade foram realizadas integrações de apenas um incremento temporal  $\Delta t$ . Do ponto de vista operacional empregou-se uma rotina que atualizava a variável de tempo inicial no arquivo *cc2d.dat* e depois executava o Featflow. Neste novo tempo inicial, a lâmina encontrava-se em novo estado, sendo a solução em pressão e velocidade consistente com o novo estado da lâmina.



## Referências

- BAXTER, L. K. *Capacitive Sensors - Design and Applications*. [S.l.]: IEEE Press Series on Electronics Technology, 1997. 302 p. ISBN 0-7803-5351-X.
- BOFFI L. V., e. a. *Conversão Eletromecânica de Energia*. EPMN 621.314.261 B633e: [s.n.], 1977.
- CÉLULAS de carga. [S.l.], 2006, Acesso em 02 de janeiro de 2006. Disponível em: <<http://www.kratoscas.com.br/telas/carga.htm>>.
- FEATFLOW - High Performance Finit Elements. [S.l.], 2007, Acesso em 12 de maio de 2007. Disponível em: <<http://www.featflow.de/dosws/dfwin.html>>.
- GMV Visualizador de elementos finitos. [S.l.], 2005, Acesso em 18 de julho de 2005. Disponível em: <<http://www-xdiv.lanl.gov/XCM/gmv/GMVHome.html>>.
- HOROVITZ, P.; HILL, W. *The Art of Electronics - 2nd ed.* [S.l.]: Cambridge University Press, 1989. 421 p. ISBN 0-521-37095-7.
- LASER, D. J.; G., S. J. A review of micropumps. *Journal of Micromechanics and Microengineering*, p. 35–64, 2003.
- LAUDER G. V., e. a. Morphology and experimental hydrodynamics of fish fin control surfaces. *Journal of Oceanic Engineering, IEEE*, v. 29, n. 3, p. 556–571, July 2004.
- LIU, P.; BOSE, N. Propulsive performance from oscillating propulsors with spanwise flexibility. *Proceedings: Mathematical, Physical and Engineering Sciences*, v. 453, p. 1763–1770, September 1997.
- MIMS, F. M. *Timer, Op Amp and Optoelectronic Circuits and Projects*. [S.l.]: Master Publishing, Inc, 2004. 124 p.
- NAKASHIMA, M.; ONO, K. Numerical study of the thrust, energy consumption, and propulsive efficiency of a three joint bending propulsion mechanisms. *Transactions of the ASME - Journal of Fluids Engineering*, v. 122, p. 614–618, September 2000.
- NAKASONE P. H., e. a. Mechatronic pump characterization: The “*mech – pump*”. *Proceedings of 18th Congress of Mechanical Engineering COBEM, Ouro Preto, MG*, p. 6–11ss, 2005.
- NÚCLEOS para bobinas. [S.l.], 2005, Acesso em 13 de fevereiro de 2005. Disponível em: <<http://www.tessin.com.br/>>.
- OGATA, K. *Discrete-Time Control Systems. 2a. Edição*. EPMN 621-50 0gld2: Prentice-Hall do Brasil, 1994. ISBN 0-13-034281-5.

- ROBOT Squid. [S.l.], 2008, Acesso em 20 de Fevereiro de 2008. Disponível em: <<http://youtube.com/watch?v=nS8iZ7ns2i4>>.
- TECHET, A. H. e. a. Boudary layer relaminarization in swimming fish. Proceedings of Ninth International Offshore and Polar Enginners Conference, p. ss, May 30-June 1999.
- TERADA, Y.; YAMAMOTO, I. An animatronic system including lifelike robotic fish. Proceedings of the IEEE. Peer Reviewed, v. 92, n. 11, p. 1814–20, November 2004.
- TRANTAFYLLOU, M. S.; TRANTAFYLLOU, G. S. An efficient swimming machine. Scientific American, p. 40–48, March 1995.
- TRANTAFYLLOU, M. S. e. a. A new paradigm of propulsion and maneuvering for marine vehicles. SNAME Transactions, v. 104, p. 81–100ss, 1996.
- TRANTAFYLLOU M. S., e. a. Review of Experimental Work in Biomimetic Foils, v. 29 No. 3, p. 585–593, July 2004.
- TUREK, S. *Efficient Solvers for Incompressible Flow Problems - An Algorithmic Approach in View of Computational Aspects*. [S.l.]: Springer, 1998.
- WALKER, J. A. Functional morphology and virtual models: Physical constrains on the design of oscillating wings, fins, legs, and feet at intermediate reynolds numbers. Integrative and Comparative Biology, v. 42, p. 232–242, 2002.
- WALKER, J. A. Kinematics and performance of maneuvering control surfaces in teleost fishes. IEEE Journal of Oceanic Engineering, v. 29, n. 3, p. 572–584, July 2004.
- WEBB, P. W. Speed, acceleration and manoeuvrability of two teleost fishes. Journal of Experimental Biology, v. 102, p. 115–122, 1983.
- WEBSTER, J. G. *The Measurement Instrumentation and Sensors Handbook*. EPEL RM 53.08(023) M463: CRC Press, 1999. ISBN 0-8493-8347-1.
- YAMAHATA C., e. a. Electromagnetically actuated ball valve. TRANSDUCERS'05 - 13th International Conference in Solid-State Sensors, Actuators and Microsystems, p. 192–196, 5-9 June 2005.
- YU, J. e. a. Development of a biomimetic robotic fish and its control algorithm. IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics, v. 34, n. 4, p. 1798–1810, August 2004.

# Livros Grátis

( <http://www.livrosgratis.com.br> )

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)  
[Baixar livros de Literatura de Cordel](#)  
[Baixar livros de Literatura Infantil](#)  
[Baixar livros de Matemática](#)  
[Baixar livros de Medicina](#)  
[Baixar livros de Medicina Veterinária](#)  
[Baixar livros de Meio Ambiente](#)  
[Baixar livros de Meteorologia](#)  
[Baixar Monografias e TCC](#)  
[Baixar livros Multidisciplinar](#)  
[Baixar livros de Música](#)  
[Baixar livros de Psicologia](#)  
[Baixar livros de Química](#)  
[Baixar livros de Saúde Coletiva](#)  
[Baixar livros de Serviço Social](#)  
[Baixar livros de Sociologia](#)  
[Baixar livros de Teologia](#)  
[Baixar livros de Trabalho](#)  
[Baixar livros de Turismo](#)