

UNIVERSIDADE FEDERAL DE MINAS GERAIS (UFMG)
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA DE ESTRUTURAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ESTRUTURAS (PROPEES)

Neimar Aparecido da Silveira Filho

**IMPLEMENTAÇÃO NÃO INTRUSIVA DO MÉTODO DOS
ELEMENTOS FINITOS GENERALIZADOS COM
ENRIQUECIMENTO GLOBAL-LOCAL**

Belo Horizonte
2023

Neimar Aparecido da Silveira Filho

**IMPLEMENTAÇÃO NÃO INTRUSIVA DO MÉTODO DOS ELEMENTOS
FINITOS GENERALIZADOS COM ENRIQUECIMENTO
GLOBAL-LOCAL**

Versão final

Dissertação apresentada ao Programa de Pós-graduação em Engenharia de Estruturas (PROPEES) da Escola de Engenharia da Universidade Federal de Minas Gerais (UFMG) como parte dos requisitos para a obtenção do título de Mestre em Engenharia de Estruturas

Orientador: Prof. Dr. Felício Bruzzi Barros

**Belo Horizonte
2023**

S587i

Silveira Filho, Neimar Aparecido da.

Implementação não intrusiva do método dos elementos finitos generalizados com enriquecimento global-local [recurso eletrônico] / Neimar Aparecido da Silveira Filho. – 2023.

1 recurso online (116 f. : il., color.) : pdf.

Orientador: Felício Bruzzi Barros.

Dissertação (mestrado) – Universidade Federal de Minas Gerais, Escola de Engenharia.

Bibliografia: f. 110-116.

Exigências do sistema: Adobe Acrobat Reader.

1. Engenharia de estruturas – Teses. 2. Método dos elementos finitos – Teses. 3. Acoplamentos – Teses. I. Barros, Felício Bruzzi.
II. Universidade Federal de Minas Gerais. Escola de Engenharia.
III. Título.

CDU: 624(043)



UNIVERSIDADE FEDERAL DE MINAS GERAIS



PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ESTRUTURAS



ATA DA DEFESA DE DISSERTAÇÃO DE MESTRADO EM ENGENHARIA DE ESTRUTURAS Nº: 396 DO ALUNO NEIMAR APARECIDO DA SILVEIRA FILHO

Às **14:00** horas do dia **06** do mês de **março** de **2023**, reuniu-se em ambiente virtual, a Comissão Examinadora indicada pelo Colegiado do Programa em **24 de fevereiro de 2023**, para julgar a defesa da Dissertação de Mestrado intitulada "**Implementação não Intrusiva do Método dos Elementos Finitos Generalizados Global-Local**", cuja aprovação é um dos requisitos para a obtenção do Grau de MESTRE EM ENGENHARIA DE ESTRUTURAS na área de Estruturas.

Abrindo a sessão, o Presidente da Comissão, **Prof. Dr. Felício Bruzzi Barros**, após dar a conhecer aos presentes o teor das Normas Regulamentares passou a palavra ao candidato para apresentação de seu trabalho. Seguiu-se a arguição pelos examinadores, com a respectiva defesa do candidato. Logo após, a Comissão se reuniu, sem a presença do candidato e do público, para julgamento e expedição do resultado final. Foram atribuídas as seguintes indicações:

Prof. Dr. Felício Bruzzi Barros - DEES - UFMG (Orientador)

Prof. Dr. Samuel Silva Penna - DEES - UFMG

Prof. Dr. Rodrigo Guerra Peixoto - DEES - UFMG

Prof. Dr. Marcos Arndt - UFPR

Pelas indicações acima, o candidato foi considerado APROVADO, conforme pareceres em anexo.

O resultado final foi comunicado publicamente ao candidato pelo Presidente da Comissão. Nada mais havendo a tratar, o Presidente encerrou a reunião e lavrou a presente ATA, que será assinada por todos os membros participantes da Comissão Examinadora.

Belo Horizonte, 06 de março de 2023.

Observações:

1. A aprovação do candidato na defesa da Dissertação de Mestrado não significa que o mesmo tenha cumprido todos os requisitos necessários para obtenção do Grau de Mestre em Engenharia de Estruturas;
2. Este documento não terá validade sem a assinatura do Coordenador do Programa de Pós-



Documento assinado eletronicamente por **Felício Bruzzi Barros, Professor do Magistério Superior**, em 06/03/2023, às 18:14, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Rodrigo Guerra Peixoto, Professor do Magistério Superior**, em 06/03/2023, às 18:14, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Samuel Silva Penna, Professor do Magistério Superior**, em 06/03/2023, às 18:15, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Marcos Arndt, Usuário Externo**, em 07/03/2023, às 10:12, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Leandro Lopes da Silva, Subcoordenador(a)**, em 17/05/2023, às 12:05, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufmg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **2106855** e o código CRC **71918002**.

Agradecimentos

Deixo agradecimentos especiais aos meus irmãos, Matheus e Izabella, pelo companheirismo, e à minha mãe, Catarina, por ser minha referência e principal apoiadora. Obrigado por fazerem tanto por mim.

Agradeço aos servidores da Universidade Federal de Minas Gerais, instituição de enorme prestígio, não somente pela qualidade de ensino, mas também pela sua importância social. Vivenciei o poder transformador da educação ao longo dos meus anos na UFMG.

Aos professores do Departamento de Engenharia de Estruturas da UFMG, agradeço pelas aulas, pelos ensinamentos e pelo profissionalismo. Em especial, ao meu orientador, professor Felício, que sempre se manteve disponível, paciente e dedicado. Sua participação foi essencial na concretização deste trabalho.

Na certeza de que nomes importantes serão esquecidos, arrisco-me a mencionar algumas pessoas pelas quais possuo enorme carinho e que, cada qual à sua maneira, foram importantes na minha vida. Aos amigos Arthur, Carol, Clara, Cibele, Edézio, Eduardo, Ellen, Ernaniriri, Laura, Lucinha, Marcelo, Saulo, Tamaris, Tarcísio e Zé, agradeço pelo suporte e pelos momentos que passamos juntos.

Às pessoas que participaram da minha trajetória na Kot Engenharia, meus sinceros agradecimentos. Os anos de convívio em um ambiente fantástico, pautado pelo incentivo à curiosidade e à colaboração, foram de fundamental importância na minha formação técnica e profissional.

Resumo

Este trabalho apresenta a implementação da estratégia de acoplamento não intrusivo para problemas estruturais multiescala denominada $MEFG^{gl}$ -GLI. Nessa abordagem, o problema é dividido em três escalas. A escala global é solucionada pelo Método dos Elementos Finitos em sua formulação convencional e não considera fenômenos localizados. A mesoescala, uma escala intermediária, e a escala local são solucionadas por meio da estratégia global-local aplicada ao Método dos Elementos Finitos Generalizados ($MEFG^{gl}$), que contempla todos os fenômenos de interesse. O acoplamento entre a escala global e a mesoescala é realizado pelo procedimento denominado Global-Local Iterativo (GLI). Um dos objetivos deste trabalho é aproximar métodos e estratégias ainda em desenvolvimento e investigação no ambiente acadêmico, como o $MEFG^{gl}$, da realidade da engenharia praticada na indústria. Nesse sentido, na implementação realizada, a escala global, cuja solução pode exigir programas computacionalmente eficientes, é resolvida pelo programa Abaqus, bastante difundido na indústria. Por sua vez, o problema composto pela mesoescala e pela escala local é resolvido na plataforma **INSANE** (*INteractive Structural ANalysis Environment*), projeto de código-aberto desenvolvido pelo Departamento de Engenharia de Estruturas da Universidade Federal de Minas Gerais. Este trabalho apresenta a validação da implementação por meio da solução de problemas estáticos com comportamento linear elástico. Em seguida, buscou-se confirmar observações disponíveis na literatura sobre o assunto, que é bastante recente. Por fim, investigações foram realizadas para avaliar a influência de alguns parâmetros do $MEFG^{gl}$ -GLI. Durante tais investigações, destaca-se que foi observado que a extensão da mesoescala influencia a taxa de convergência do GLI. A avaliação de técnicas de aceleração de convergência indicou que o uso da relaxação dinâmica evita uma das principais limitações do método, que reside no fato de a taxa de convergência ser predeterminada pela diferença de rigidez entre as escalas acopladas.

Palavras-chave: Método dos Elementos Finitos Generalizados. Análise Global-Local. Acoplamento não intrusivo.

Abstract

This work implements a non-intrusive coupling strategy for multiscale structural problems known as IGL-GFEM^{gl}. In this approach, the problem is divided into three scales. The global scale is solved by standard Finite Element Method (FEM) and it does not evaluate any local feature. The mesoscale, which is an intermediary scale, and the local scale compose a second problem which is addressed by the global-local strategy applied to the Generalized Finite Element Method (GFEM^{gl}). All the local features are properly modeled on the local problem. The compatibility between global and mesoscale solutions is assured by the Iterative Global-Local (IGL), a non-intrusive strategy for problem coupling. This work aims to bring methods and resources still under development in the academia, such as GFEM^{gl}, closer to the industry reality. Regarding that, in the present implementation, the global scale, which often requires computational efficiency, is solved by Abaqus, a widespread software in the industry. On the other hand, meso and local scales are dealt by the **INSANE** platform (INteractive Structural ANalysis Environment), an open-source project developed by the Department of Structural Engineering of the Federal University of Minas Gerais. The implementation is validated by a set of static elastic linear problems. Then, some observations available in the related literature were explored. Finally, a set of simulations were evaluated in order to investigate some parameters of the IGL-GFEM^{gl}. The results indicate that the size of the mesoscale impacts the convergence rate of the GLI algorithm. It was also shown that the dynamic relaxation overcomes a major limitation of the method. By the use of this technique, the initial stiffness gap between the scales no longer controls the convergence rate of the solution.

Key-words: Generalized Finite Element Method. Global-Local Analysis. Non-intrusive coupling.

Lista de Figuras

2.1	Nuvem de círculos (ou esferas) centradas nos nós.	25
2.2	Partição da Unidade para elementos finitos em \mathbb{R}^1	27
2.3	Processo de enriquecimento de uma PU.	29
2.4	Flexibilidade nas funções de forma obtidas pelo MEFG.	29
2.5	Procedimento de solução pelo MEFG ^{gl}	31
2.6	Domínios global e local no MEFG ^{gl}	32
2.7	Problema de decomposição de domínio.	38
2.8	Problema global-local.	41
2.9	Solução de referência obtida pelo acoplamento não intrusivo para análise global-local.	43
2.10	Processo de solução pelo MEF-GLI.	46
2.11	Processo de solução pelo MEFG ^{gl} -GLI.	52
3.1	Identificação dos nós e dos elementos de interface no MEFG ^{gl} -GLI.	55
3.2	Problema geral de um acoplamento não intrusivo.	58
4.1	Problema 1 – Barra submetida a força concentrada em uma das extremi- dades.	78
4.2	Forças residuais no MEF-GLI – Ω_L mais rígido (azul) e menos rígido (ver- melho) no modelo local.	79
4.3	Taxa de convergência típica do MEF-GLI para problemas lineares elásticos.	79
4.4	Solução de referência para o Caso 2 do Problema 1.	80
4.5	Problema 2 – Chapa com orifício central sob tração pura.	82
4.6	Deslocamentos na direção y (u_y) obtidos pelo MEFG ^{gl} -GLI.	83
4.7	Deslocamentos na direção y (u_y) obtidos pelo Modelo A.	84
4.8	Deslocamentos na direção y (u_y) obtidos pelo Modelo B.	84
4.9	Comparação dos deslocamentos u_y na borda inferior da chapa obtidos pelo MEFG ^{gl} -GLI e pelos Modelos A e B.	85
4.10	Componente de deformação ε_{yy} na mesoescala. Resultados obtidos pelo MEFG ^{gl} -GLI.	87
4.11	Componente de deformação ε_{yy} na mesoescala. Resultados obtidos pelo MEF.	87

4.12	Definição dos pontos 1 e 2 para comparação de resultados.	88
4.13	Componente de deformação ε_{xx} nos pontos 1 e 2 em função da tolerância adotada.	89
4.14	Tempo de execução para diferentes valores de tolerância.	90
4.15	Componente de deformação ε_{xx} na mesoescala. Resultados obtidos pelo MEF gl -GLI.	91
4.16	Componente de deformação ε_{xx} na mesoescala. Resultados obtidos pelo MEF.	91
4.17	Tempo de execução para diferentes n_{GL}	93
4.18	Modelos utilizados na avaliação da influência da abrangência da mesoescala nos resultados do MEF gl -GLI.	94
4.19	Resultados dos Modelos M1, M2 e M3.	95
4.20	Convergência dos Modelos M1, M2 e M3.	96
4.21	Convergência utilizando relaxação estática com diferentes valores de ω	98
4.22	Relação entre o número de iterações MEF gl -GLI e ω	98
4.23	Deslocamentos e deformações obtidas pelo MEF gl -GLI utilizando relaxação estática com $\omega = 0,35$	100
4.24	Deslocamentos e deformações obtidas pelo Modelo B.	100
4.25	Deslocamentos verticais na borda inferior da chapa pelo Modelo B e pelo MEF gl -GLI com $\omega = 0,35$	101
4.26	Deslocamentos u_x em $x = 45,0$ u.c. obtidos pelo Modelo B (MEF) e pelo MEF gl -GLI com $\omega = 0,35$	102
4.27	Convergência do MEF gl -GLI utilizando diferentes técnicas de aceleração.	103
4.28	Deslocamentos verticais na borda inferior da chapa obtidos pelo Modelo B (MEF) e o e pelo MEF gl -GLI com $\omega = 0,35$ e com relaxação dinâmica.	104
4.29	Convergência utilizando relaxação dinâmica para diferentes rigidezes da mesoescala.	105

Lista de Tabelas

3.1	Códigos responsáveis pelas etapas do $MEFG^{gl}$ -GLI.	56
3.2	Principais atributos da classe <code>coupler</code>	70
3.3	Principais atributos da classe <code>model</code>	72
3.4	Principais atributos das classes <code>accelToolBox</code> , <code>staticRelax</code> , <code>dynamicRelax</code> e <code>quasiNewtonAccel</code>	75
4.1	Comparação de resultados entre o MEF e o MEF-GLI.	80
4.2	Comparação de resultados do $MEFG^{gl}$ -GLI para diferentes valores de to- lerância (ϵ).	88
4.3	Custo computacional do $MEFG^{gl}$ -GLI para diferentes valores de tolerância (ϵ).	89
4.4	Resultados do $MEFG^{gl}$ -GLI para diferentes números de ciclos $MEFG^{gl}$. . .	92
4.5	Custo computacional do $MEFG^{gl}$ -GLI para diferentes números de ciclos $MEFG^{gl}$	92
4.6	Convergência do $MEFG^{gl}$ -GLI para diferentes números ciclos $MEFG^{gl}$. . .	93
4.7	Custo computacional do $MEFG^{gl}$ -GLI para diferentes tamanhos da meso- escala.	96
4.8	Resultados do $MEFG^{gl}$ -GLI para alguns valores de ω	99
4.9	Fator de relaxação por iteração para diferentes valores de $n = E_M/E_G$. . .	104

Lista de abreviaturas e siglas

GLI	Global-Local Iterativo
HNA	<i>Hierarchical Non-Intrusive Algorithm</i>
IGL	<i>Iterative Global-Local</i>
IGL-FEM	<i>Iterative Global-Local Finite Element analysis</i>
INSANE	<i>INteractive Structural ANalysis Environment</i>
MEF	Método dos Elementos Finitos
MEF-GLI	Método dos Elementos Finitos Global-Local Iterativo
MEFG	Método dos Elementos Finitos Generalizados
MEFGE	Método dos Elementos Finitos Generalizados Estável
MEFG ^{gl}	Método dos Elementos Finitos Generalizados com enriquecimentos Global-Local
MEFG ^{gl} -GLI	Estratégia Global-Local Iterativa aplicada ao Método dos Elementos Finitos Generalizados com enriquecimentos Global-Local
MMQM	Método dos Mínimos Quadrados Móveis
PU	Partição da Unidade
POO	Programação Orientada à Objetos
XFEM	<i>eXtended Finite Element Method</i>
u.c.	Unidades consistentes

Lista de Símbolos

Sobrescritos

- \cdot^0 Índice que se refere a graus de liberdade do problema global inicial
- \cdot^D Índice relativo à condições de contorno de Dirichlet
- \cdot^N Índice relativo à condições de contorno de Neumann
- \cdot^{gl} Índice que se refere a graus de liberdade introduzidos pelo enriquecimento
- \cdot^k Índice relativo à iteração do GLI ou do MEF G^{gl}
- \cdot^Γ Índice relativo à interface entre domínios

Subscritos

- \cdot_1 Índice relativo ao domínio 1 ($\bar{\Omega}_1$)
- \cdot_2 Índice relativo ao domínio 2 ($\bar{\Omega}_2$)
- \cdot_C Índice relativo ao domínio complementar ($\bar{\Omega}_C$)
- \cdot_G Índice relativo ao domínio global ($\bar{\Omega}_G$)
- \cdot_{GC} Índice relativo ao domínio complementar no problema global
- \cdot_{GL} Índice relativo ao domínio local no problema global
- \cdot_L Índice relativo ao domínio local ($\bar{\Omega}_L$)
- \cdot_M Índice relativo ao domínio da mesoescala
- \cdot_{ME} Índice relativo ao domínio da mesoescala enriquecida
- \cdot_i Índice relativo às funções de forma
- \cdot_j Índice relativo aos nós
- \cdot_ψ Índice relativo ao espaço dos multiplicadores de Lagrange ($L^2(\Gamma)$)

Letras maiúsculas

A	Matriz de acoplamento
C	Tensor de Hooke
E	Módulo de elasticidade
H^1	Espaço de Hilbert de ordem 1
L	Função de aproximação local ou função de enriquecimento
K	Matriz de rigidez
N	Função básica da PU
Q_N	Conjunto de N nós
R	Vetor de forças residuais
S	Complemento de Schur de uma matriz
X	Espaço de discretização

Letras minúsculas

<i>a</i>	Forma bilinear associada a um domínio
<i>b</i>	Forma bilinear associada a Γ
b	Parâmetro associado a graus de liberdade do enriquecimento nodal
f	Vetor de forças
f_B	Vetor de forças de corpo
<i>l</i>	Forma linear associada às condições de contorno naturais
\hat{n}	Vetor unitário normal à uma superfície
n_{GL}	Número de ciclos global-local em uma solução pelo MEF gl
p	Funções teste associadas ao espaço dos multiplicadores de Lagrange
q_j	Número de funções de forma associadas ao nó \mathbf{x}_j
r_j	Dimensão da nuvem ω_j
t	Vetor de forças de superfície
$\tilde{\mathbf{t}}$	Vetor de forças de superfície prescritas
u	Solução numérica aproximada
$\tilde{\mathbf{u}}$	Vetor de deslocamentos prescritos
$\bar{\mathbf{u}}$	Solução global prevista (não corrigida)
\mathbf{u}^{MEF}	Solução numérica associada à formulação convencional do MEF

\mathbf{u}^{enr}	Solução numérica associada ao enriquecimento da PU
\mathbf{u}_G^0	Solução global inicial
\mathbf{u}_j	Deslocamentos associados ao nó \mathbf{x}_j
\mathbf{u}_{ref}	Solução numérica de referência para o GLI
u_x	Componente de deslocamento na direção x
u_y	Componente de deslocamento na direção y
\mathbf{v}	Funções teste associadas ao espaço de solução do problema
\mathbf{x}	Posição em um domínio
\mathbf{x}_j	Nó em um domínio

Letras gregas maiúsculas

Γ	Interface entre domínios
$\Delta \mathbf{u}$	Incremento da solução numérica \mathbf{u}
Δ^k	Vetor de incremento de deslocamentos de Aitken
Ω	Domínio aberto e suave
$\partial\Omega$	Contorno do domínio Ω
$\bar{\Omega}$	Domínio fechado

Letras gregas minúsculas

ϵ	Tolerância para o resíduo relativo no GLI ou MEF G^{gl} -GLI
$\boldsymbol{\varepsilon}$	Tensor de deformações
ε_{xx}	Componente de deformação normal ao plano yz na direção x
ε_{xy}	Componente de deformação normal ao plano yz na direção y
ε_{yy}	Componente de deformação normal ao plano xz na direção y
$\boldsymbol{\eta}$	Vetor de forças em Γ
κ	Parâmetro de rigidez de Cauchy
ν	Coefficiente de Poisson
$\boldsymbol{\sigma}$	Tensor de Cauchy
φ	Função de forma
$\boldsymbol{\psi}$	Multiplicadores de Lagrange
ω	Fator de relaxação

- ω^k Fator de relaxação na iteração k
 ω_j Nuvem associada ao nó \mathbf{x}_j
 ω_{otm} Valor ótimo para o fator de relaxação

Operadores e símbolos especiais

- ∇ Operador nabla
 $\|\mathbf{a}\|$ Norma euclidiana do vetor \mathbf{a}
 $\underline{\mathbf{a}}$ Prolongamento do vetor ou da matriz \mathbf{a}
 $(e)_j$ Elemento na posição j de um vetor
 $(e)_{ij}$ Elemento na linha i e coluna j de uma matriz

Conjuntos numéricos

- \mathbb{R} Conjunto dos números reais
 \mathbb{Z} Conjunto dos números inteiros

Sumário

1	Introdução	18
1.1	Justificativa	20
1.2	Objetivos	21
1.2.1	Objetivo geral	21
1.2.2	Objetivos específicos	21
1.3	Metodologia	22
1.4	Organização do texto	22
2	Revisão Bibliográfica	24
2.1	Método dos Elementos Finitos Generalizados	24
2.1.1	Formulação	27
2.2	Método dos Elementos Finitos Generalizados com enriquecimento global- local	30
2.2.1	Formulação	32
2.2.2	Considerações sobre o $MEFG^{gl}$	34
2.3	Estratégias de acoplamento	36
2.3.1	Problema de decomposição do domínio	38
2.3.2	Acoplamento na estratégia global-local	41
2.3.3	Acoplamento não intrusivo para análise global-local iterativa	43
2.4	Método dos Elementos Finitos Global-Local Iterativo	45
2.5	Acoplamento não intrusivo para o $MEFG^{gl}$	49
2.5.1	$MEFG^{gl}$ e algoritmo Global-Local Iterativo	50
3	Aspectos computacionais do $MEFG^{gl}$-GLI	53
3.1	Algoritmo	53
3.2	Convergência do GLI	56
3.2.1	Forma incremental do problema global	57
3.2.2	Comportamento da convergência	59
3.3	Técnicas de aceleração	61
3.3.1	Relaxação estática	61
3.3.2	Relaxação Dinâmica	62

3.3.3	Quasi-Newton	64
3.4	Implementação computacional	66
3.4.1	Características gerais	67
3.4.2	Requisitos	69
3.4.3	Classe <i>coupler</i>	69
3.4.4	Classe <i>model</i>	71
3.4.5	Classe <i>accelToolBox</i>	73
3.4.6	Classe <i>linAlg</i>	76
4	Simulações numéricas	77
4.1	Solução pelo MEF-GLI	77
4.2	Validação do MEF G^{gl} -GLI no INSANE	81
4.3	Validação do MEF G^{gl} -GLI no Abaqus- INSANE	84
4.3.1	Influência da tolerância	85
4.3.2	Influência do número de ciclos MEF G^{gl}	90
4.3.3	Influência da extensão da mesoescala	94
4.3.4	Técnicas de aceleração	96
5	Considerações finais	106
5.1	Contribuições	106
5.2	Sugestões para trabalhos futuros	108
	Bibliografia	110

Capítulo 1

Introdução

Problemas que envolvem singularidades, descontinuidades e elevados gradientes de deformação são comuns na prática da engenharia. Esses problemas representam um desafio para o Método dos Elementos Finitos (MEF) em sua formulação convencional. Sua aplicação direta pode ser inviável ou resultar em erros significativos na solução aproximada obtida (Belytschko et al., 2009).

O Método dos Elementos Finitos Generalizados (MEFG) apresenta-se, então, como uma alternativa. Adotando as funções de forma do MEF para a aplicação do conceito de Partição da Unidade (PU), o MEFG permite a construção de funções de forma enriquecidas por funções de aproximação local (Strouboulis et al., 2000). Segundo Barros (2002), tais funções, também conhecidas como funções de enriquecimento, são capazes de capturar comportamentos localizados como singularidades e descontinuidades. Essas características reduzem a necessidade da realização de refinamento h para que a modelagem do problema seja realizada de maneira adequada (Belytschko et al., 2009). Exige-se, contudo, o conhecimento prévio do comportamento da solução para que seja selecionada uma função de enriquecimento adequada, isto é, que conduza a bons resultados.

Conforme Pinheiro (2019), o Método dos Elementos Finitos Generalizados com funções de enriquecimento global-local (MEFG^{gl}) consiste em uma estratégia para a construção de funções de enriquecimento para um problema, chamado de problema global, a partir da solução de problemas locais estabelecidos em sub-regiões do domínio de análise. A principal vantagem dessa abordagem está em se construir as funções de enriquecimento sem a necessidade do conhecimento prévio do comportamento do problema. Sua aplicação é justificada sempre que algum fenômeno de interesse estiver confinado a uma pequena região do problema — descontinuidades e elevados gradientes típicos de problemas com trinca, por exemplo. Desse modo, soluções de problemas complexos, cujo comportamento local é de difícil representação, podem ser obtidas sem a necessidade de se grandes modelos com malha extremamente refinada nas regiões com comportamento localizado. Em vez disso, são realizadas análises locais, cujas aproximações obtidas numericamente enriquecem a solução aproximada do problema global.

Recentemente, programas comerciais como o ANSYS e o Abaqus implementaram o *eXtended Finite Element Method* (XFEM), proposto por Belytschko e Black (1999), para a solução de problemas que envolvem a mecânica da fratura. Belytschko, Gracie e Ventura (2009) esclareceram que o XFEM e o MEF G^{gl} são considerados métodos equivalentes. Já o MEF G^{gl} , por sua vez, permanece restrito a programas desenvolvidos em ambiente acadêmico. Uma forma de se viabilizar a aplicação do MEF G^{gl} na indústria consiste em utilizar algoritmos para o acoplamento dessa estratégia com métodos já implementados nos programas comerciais.

Vários métodos de acoplamento foram desenvolvidos para permitir que diferentes estratégias sejam utilizadas em conjunto para a solução de problemas de engenharia (Duval et al., 2014). Implementações não intrusivas têm estado em evidência em pesquisas recentes e podem ser definidas como as estratégias de acoplamento cuja implementação dispensa a modificação dos *software* ou das estratégias de solução envolvidas (*solvers*). Dentre as diversas propostas de acoplamento não intrusivo, podem ser citadas: abordagem global-local multidimensional (elementos 2D na escala global e elementos 3D na escala local) (Guguin et al., 2014); problemas dinâmicos em regime transiente (Bettinotti et al., 2014); estudo de propagação de trincas (Gupta, Pereira, Kim, Duarte e Eason, 2012) e análise de gradientes térmicos (Plews et al., 2012). Destaca-se, ainda, o trabalho de Fillmore e Duarte (2018) sobre algoritmos de acoplamento hierárquico não intrusivo para o MEF G .

Neste trabalho, realizou-se uma implementação não intrusiva do MEF G^{gl} para a análise de problemas multiescala¹ conforme proposto por Li, O'Hara e Duarte (2021). A escala global é solucionada em programa comercial por meio do MEF. Para esse propósito, foi utilizado o Abaqus. A escala intermediária, chamada mesoescala, e a escala local são solucionadas pelo MEF G^{gl} através plataforma **INSANE**. O acoplamento entre a escala global e a mesoescala é não intrusivo e, para isso, é utilizada a estratégia proposta por Whitcomb (1991) denominada *Iterative Global-Local* (IGL ou, no idioma deste texto, Global-Local Iterativo, GLI). Será, portanto, necessário compatibilizar e transferir as informações entre os pacotes de *software*, já que, por ser não intrusiva, a implementação não contempla alterações no código-fonte de nenhuma das aplicações envolvidas.

O sistema computacional *INteractive Structural ANalysis Environment* (**INSANE**) é um projeto de *software* livre em constante desenvolvimento, criado e mantido pelo Departamento de Engenharia de Estruturas da Universidade Federal de Minas Gerais (Fuina, 2004, Penna, 2007). É escrito na linguagem JAVA e, assim, adota o paradigma da Programação Orientada a Objetos. O núcleo numérico do projeto inclui classes voltadas à implementação do MEF G e do MEF G^{gl} (Alves, 2012, Malekan, 2017, Fonseca, 2019).

¹No contexto deste texto, o termo multiescala se refere à abordagem na qual um problema é resolvido por meio de duas ou mais escalas de solução. Isto é, o termo indica que foram construídos modelos computacionais com diferentes escalas de precisão e abrangência.

1.1 Justificativa

Com frequência, as estruturas de engenharia apresentam regiões que demandam análise mais minuciosa devido à ocorrência de fenômenos localizados. Em geral, esses fenômenos são mais recorrentes em partes específicas das estruturas, por exemplo, onde ocorrem descontinuidades — trincas, aberturas e juntas — e onde há elevados gradientes de deformação — detalhes geométricos, regiões de apoio, locais de introdução de ações estruturais e contato estrutural.

A solução numérica de problemas localizados são, em geral, onerosas em termos do tempo de modelagem e de processamento. Com frequência, a obtenção da solução local requer que a região do entorno seja considerada na análise. Ainda, também há o interesse em se investigar as consequências dos efeitos localizados no comportamento global da estrutura. Isto é, o comportamento global pode ser importante para o estudo de efeitos locais, cuja solução pode alterar o comportamento em escala global. Por exemplo, o estudo da propagação de uma trinca em uma viga de um edifício requer a avaliação da resposta de toda a estrutura mediante às ações que nela atuam. Por outro lado, os resultados da propagação da trinca podem indicar uma redistribuição de esforços no edifício.

Nesse cenário, pode-se pensar em uma abordagem na qual o problema seja considerado por meio de um único modelo computacional que considere toda a estrutura e os fenômenos locais de interesse. Existem casos nos quais tal abordagem é aplicável. Na maioria das situações que ocorrem na prática da engenharia, entretanto, isso não ocorre por vários motivos. Há dificuldade de se manipular modelos computacionais muito grandes, isto é, mesmo os programas computacionais voltados para a indústria possuem suas limitações. Também é mais difícil conferir e validar modelos muito extensos. A construção de um modelo único também causa limitações no processo de modelagem, uma vez que deve-se escolher entre utilizar o mesmo tipo de elemento finito em toda a estrutura ou realizar o acoplamento de diferentes tipos de elementos. Por fim, ainda que todas essas limitações sejam contornadas, a solução do problema resultante pode ter custo computacional elevado, talvez até impeditivo.

Na prática, é comum que as estruturas de grande porte sejam modeladas com o objetivo de capturar comportamentos globais, enquanto os efeitos localizados, se presentes, são avaliados com o uso de metodologias analíticas, quando disponíveis, ou por meio de modelos numéricos à parte. Essa atividade requer conhecimento sobre quais fenômenos localizados podem ser relevantes para a estrutura, já que o modelo estrutural global não irá, necessariamente, capturar tais efeitos. No contexto dessa abordagem, voltando-se ao exemplo da propagação de trinca em uma viga de um edifício, o comportamento global do edifício seria capturado por meio de um modelo mais simples, isto é, que ignore a propagação da trinca. Essa resposta seria, então, utilizada na construção de um segundo modelo, elaborado de modo a considerar a viga com todos os seus efeitos locais relevantes.

Finalmente, esses resultados são utilizados para avaliar as repercussões da propagação da trinca no comportamento global da estrutura.

Há, assim, interesse em métodos e estratégias que permitam a construção e a análise de problemas locais de modo eficiente e que levem em consideração a influência do comportamento global da estrutura. O desenvolvimento dessas técnicas é recente e ainda restrito ao ambiente acadêmico, de modo que elas ainda não se encontram disponíveis nos pacotes de *software* utilizados em larga escala.

No contexto atual, o acoplamento não intrusivo para análise global-local é uma solução interessante, permitindo o uso conjunto da robustez de um *software* comercial com os recursos ainda em desenvolvimento e consolidação em meio acadêmico. Por exemplo, a plataforma computacional **INSANE** implementa o MEFG, o MEFG^{gl} e o MEFGE (Método dos Elementos Finitos Generalizados Estável), porém, apresenta limitações de desempenho que dificultam a sua aplicação na solução de problemas de larga escala. O acoplamento não intrusivo, portanto, permite a aplicação dos métodos implementados no **INSANE** em conjunto com um *software* capaz de realizar soluções pelo MEF com alta eficiência computacional.

Ainda que já existam metodologias para a implementação de acoplamentos não intrusivos, o tema consiste em uma linha de pesquisa recente. Muitos dos parâmetros envolvidos ainda não foram investigados, bem como ainda é necessário avaliar o desempenho das metodologias disponíveis quando aplicadas à diferentes classes de problemas. Nesse sentido, essa é uma área que se encontra em desenvolvimento e em investigação.

1.2 Objetivos

1.2.1 Objetivo geral

Este trabalho tem como objetivo geral contribuir para aproximar métodos e ferramentas disponíveis na academia da realidade exigida pela indústria. Para isso, busca-se implementar o acoplamento não intrusivo para a análise de problemas multiescala pelo MEFG^{gl}. A escala global é avaliada pelo MEF através de um programa comercial, enquanto os efeitos localizados são analisados por meio do MEFG^{gl}. A metodologia foi proposta por Li, O'Hara e Duarte (2021) e é denominada MEFG^{gl}-GLI por utilizar o GLI para realizar o acoplamento entre os resultados do MEF e do MEFG^{gl}. No contexto deste trabalho, a implementação realizada avalia a escala global por meio do *software* Abaqus, enquanto o **INSANE** foi utilizado para a análise pelo MEFG^{gl}.

1.2.2 Objetivos específicos

Este trabalho possui os seguintes objetivos específicos:

- i. Possibilitar o uso conjunto de recursos disponíveis em um programa de pesquisa (**INSANE**) com um programa voltado para a indústria (Abaqus).
- ii. A partir da implementação do MEFG^{gl} já disponível no **INSANE**, investigar o desempenho da estratégia de solução MEFG^{gl} -GLI.
- iii. Comparar os resultados obtidos por meio do MEFG^{gl} -GLI e do MEF convencional, de modo a avaliar a qualidade dos resultados e a demanda computacional.

1.3 Metodologia

A seguinte metodologia foi adotada neste trabalho:

- i. Revisão bibliográfica sobre o MEFG^{gl} e sobre os algoritmos de acoplamento não intrusivo no escopo da análise global-local.
- ii. Familiarização com a implementação do MEFG^{gl} na plataforma **INSANE**.
- iii. Implementação do acoplamento não intrusivo para a solução de problemas multi-escala por meio do MEFG^{gl} . Foi utilizada a estratégia proposta por Li, O'Hara e Duarte (2021). A implementação realizada possui as seguintes características:
 - a. Realiza o acoplamento entre a escala global e a mesoescala de forma não intrusiva.
 - b. Permite que as iterações entre as soluções da escala global e da mesoescala sejam realizadas de forma autônoma, isto é, sem a intervenção por parte do usuário.
 - c. Disponibiliza técnicas de aceleração da convergência.
 - d. Apresenta a solução final de forma apropriada.
- iv. Seleção de problemas de referência para a validação da implementação do acoplamento não intrusivo.
- v. Validação dos recursos implementados e realização das eventuais correções de código.
- vi. Realização de investigações acerca da qualidade dos resultados, do desempenho computacional e do comportamento da convergência do MEFG^{gl} -GLI. Também foram avaliadas as técnicas de convergência implementadas.

1.4 Organização do texto

Este texto busca documentar o trabalho desenvolvido e está organizado em 5 capítulos. O presente capítulo introduz o tema, bem como apresenta uma justificativa para a realização deste trabalho. Também contém os objetivos gerais e específicos e a metodologia aplicada.

O segundo capítulo busca apresentar uma revisão bibliográfica dos conceitos que foram necessários para o desenvolvimento do trabalho. Nesse sentido, o Capítulo 2 aborda o Método dos Elementos Finitos Generalizados (MEFG), a sua variação obtida a partir da estratégia de enriquecimento global-local (MEFG^{gl}), uma visão geral sobre as estratégias de acoplamento, o método Global-Local Iterativo para o MEF (MEF-GLI) e, por fim, o método que associa o MEFG^{gl} e o GLI (MEFG^{gl}-GLI).

O Capítulo 3 está reservado aos aspectos computacionais relacionados ao trabalho. Nele, estão apresentados os algoritmos utilizados, bem como as técnicas para a aceleração de tais algoritmos. Também são apresentadas informações sobre o código elaborado, com descrições e comentários sobre as classes e os métodos que o constituem.

No penúltimo capítulo são apresentadas simulações numéricas. As simulações tem como objetivo validar a implementação realizada, além de permitir a condução de investigações em relação ao MEFG^{gl}-GLI.

O capítulo de encerramento consiste em considerações finais. É dado destaque às principais contribuições deste trabalho, bem como são apresentadas sugestões para trabalhos futuros relacionados ao tema.

Capítulo 2

Revisão Bibliográfica

Este capítulo apresenta uma revisão bibliográfica sobre os assuntos relacionados ao tema deste texto. A formulação do Método dos Elementos Finitos Generalizados é apresentada. Explora-se, então, a estratégia baseada na construção de funções de enriquecimento global-local para o MEFG, denominada $MEFG^{gl}$. Em seguida, apresenta-se breve revisão sobre estratégias de acoplamento para problemas com decomposição de domínio, com destaque para o Método dos Elementos Finitos Global-Local Iterativo, MEF-GLI. Finalmente, todos os conceitos apresentados são utilizados para a apresentação do $MEFG^{gl}$ -GLI, cuja implementação computacional consiste no tema deste trabalho.

2.1 Método dos Elementos Finitos Generalizados

Conforme Barros (2002), o Método dos Elementos Finitos Generalizados (MEFG) pode ser interpretado como uma variação do MEF que compartilha diversas características com os métodos sem malha. Duarte, Babuška e Oden (2000) afirmam que a formulação do MEFG tem origem em dois trabalhos desenvolvidos de forma independente:

- Babuška e colegas desenvolveram o “Método dos Elementos Finitos Especiais” (Babuška, Caloz e Osborn, 1994) e, posteriormente, o Método da Partição da Unidade (Melenk e Babuška, 1996).
- Duarte e Oden (1995b) formularam o Método das Nuvens hp e, posteriormente, apresentaram uma abordagem híbrida do MEF com métodos sem malha (Oden, Duarte e Zienkiewicz, 1996).

Nos Métodos sem Malha, a solução aproximada é obtida sem que uma malha de elementos seja necessária. Em vez disso, é utilizado um conjunto de nós dispostos ao longo do domínio de interesse sem conectividade fixa entre si. Essa classe de métodos, portanto, busca obter soluções aproximadas para problemas de valor de contorno sem a necessidade do particionamento explícito do domínio, como é realizado, por exemplo, no

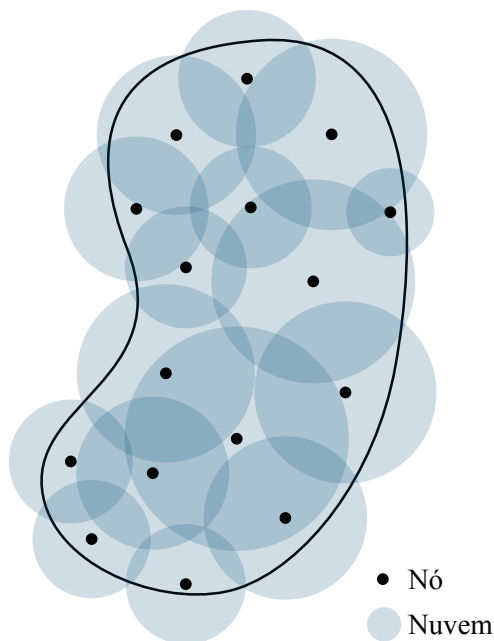


Figura 2.1: Nuvem de círculos (ou esferas) centradas nos nós.
Adaptado de Duarte e Oden (1995a).

Método dos Elementos Finitos. Duarte e Oden (1995a) apresentam uma revisão dessa classe de métodos.

O Método das Nuvens hp , já mencionado, pertence a tal classe de métodos. A aproximação é obtida com base na distribuição de nós sem conectividade preestabelecida e é enriquecida pela inclusão de funções locais de aproximação. O método foi proposto com o objetivo de apresentar boa eficiência computacional enquanto mantém rigor matemático. A seguir, uma breve explicação sobre o método é apresentada para, na sequência, derivar o MEFG.

Seja um conjunto de nós arbitrariamente definidos $\mathbf{Q}_N = \{\mathbf{x}_j\}_{j=1}^N$, $\mathbf{x}_j \in \Omega$, em que Ω é um domínio aberto e suave. Uma região do domínio Ω é, então, definida pela composição das N nuvens ω_j centradas nos nós do conjunto \mathbf{Q}_N , conforme Fig. 2.1. A nuvem ω_j é definida por

$$\omega_j := \mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x} - \mathbf{x}_j\| \leq r_j, \quad (2.1)$$

em que r_j é a dimensão do espaço real \mathbb{R} é dado por $n = 1, 2$ ou 3 . Desse modo, tem-se que o domínio fechado $\bar{\Omega}$ está contido no espaço $\cup_{j=1}^N \{\omega_j\}$, isto é, $\bar{\Omega} \subset \cup_{j=1}^N \{\omega_j\}$.

As funções de aproximação do Método das Nuvens hp são obtidas a partir da construção da Partição da Unidade (PU). Conforme Duarte e Oden (1995a), a PU é um conjunto de funções $\{\varphi_j\}_{j=1}^N$ tais que apresentem as seguintes propriedades:

Propriedade 1 $\varphi_j \in C_0^\infty(\omega_j)$, $1 \leq j \leq N$.

Propriedade 2 $\sum_{j=1}^N \varphi_j(\mathbf{x}) = 1$, $\forall \mathbf{x} \in \bar{\Omega}$.

Barros (2002) destaca que as funções obtidas pelo Método dos Mínimos Quadrados

Móveis (MMQM) (Lancaster e Salkauskas, 1981) e as funções de Shepard (Shepard, 1968) são exemplos de conjuntos de funções capazes de construir uma PU. Nota-se que as funções Lagrangianas, frequentemente utilizadas nas aproximações do MEF, atendem estritamente apenas à Propriedade 2. Ainda assim, podem ser consideradas PU, dentro de um contexto em que o requisito de atendimento à Propriedade 1 é flexibilizado.

Segundo Duarte e Oden (1996), a integração numérica do sistema de equações obtido pelo Método das Nuvens *hp* é computacionalmente onerosa em comparação ao MEF convencional. Além disso, conforme Barros (2002), há instabilidade numérica devido à imposição de condições de contorno essenciais (Dirichlet) por meio da formulação fraca do problema. Essa limitação pode ser contornada pelo uso do Método dos Multiplicadores de Lagrange (Belytschko et al., 1994), que, por sua vez, também pode ser computacionalmente oneroso.

A estratégia adotada pelo MEFG é, então, desenhada com o objetivo de resolver os problemas observados no Método das Nuvens *hp*. Isso é, em grande parte, feito com o uso da malha e das funções de forma do MEF para a construção da PU. Barros (2002) indica que essa medida resulta em redução do custo da integração numérica, como também reduz a instabilidade numérica por viabilizar a imposição direta das condições de contorno essenciais (Dirichlet), desde que as funções de enriquecimento (comentadas a seguir e descritas na Seção 2.1.1) sejam adequadamente escolhidas. Outra vantagem no uso das funções de forma do MEF reside na simplicidade da sua implementação. Os problemas numéricos associados aos MEF, contudo, também são herdados. Por exemplo, a adoção de elementos muito distorcidos prejudica, ainda que em menor grau, a qualidade da aproximação.

No MEFG, tal como no Método das Nuvens *hp*, o enriquecimento da PU é realizado multiplicando-a por um conjunto de funções linearmente independentes. Tal enriquecimento possui papel importante na qualidade da aproximação obtida, pois, segundo Barros (2002), permite a obtenção de aproximações adequadas para problemas cuja solução não é satisfatoriamente representada pelas funções de forma do MEF e que isso é feito sem a necessidade de se introduzir novos pontos de discretização ao domínio.

A principal vantagem associada ao uso do MEFG está, portanto, associada à ocorrência de fenômenos que desafiam as limitações do MEF. Ainda que largamente difundido e relativamente simples de ser utilizado, a formulação convencional do MEF não é eficiente para a solução de algumas classes de problemas relevantes, tais quais os que envolvem descontinuidades, vazios, singularidades e elevados gradientes de deformação. Seja por facilitar o processo de modelagem ou por permitir uma solução de melhor qualidade, o MEFG pode ser utilizado nos casos em que o MEF não é eficiente, ou quando oferece uma solução pobre. Barros (2002) ainda comenta que o MEFG confere maior flexibilidade para a obtenção de soluções numéricas aproximadas para problemas de valor de contorno, conforme será discutido na Seção 2.1.1.

Há, ainda, outro método derivado do MEF que compartilha características dos métodos sem malha e que foi desenvolvido em paralelo ao processo de consolidação da formulação do MEFG. Sob o nome Método dos Elementos Finitos Estendido (XFEM, do inglês *eXtended Finite Element Method*) e proposto por Belytschko e Black (1999), esse método tinha como objetivo o enriquecimento de apenas parte dos nós que constituem o modelo computacional, ao passo que o MEFG, em sua origem, considerava o enriquecimento de todos os nós. O XFEM notabilizou-se pelo uso de um conjunto de funções de enriquecimento – conjunto este distinto daquele adotado nas formulações do MEFG – para se descrever a singularidade do campo de tensões na vizinhança da ponta da trinca. Adicionalmente, foi no XFEM que as funções de salto foram primeiramente introduzidas como funções de enriquecimento (Moës, Dolbow e Belytschko, 1990). As funções de salto são utilizadas para descrever a descontinuidade advinda da presença da trinca.

Com o avanço das pesquisas, o MEFG e o XFEM convergiram e o consenso atual é que consistem em nomenclaturas distintas para o mesmo método (Belytschko, Gracie e Ventura, 2009). Assim, muitos trabalhos referem-se ao método como MEFG/X (ou, em inglês, G/XFEM). Neste trabalho, adotou-se, por simplicidade, a grafia MEFG.

2.1.1 Formulação

No MEFG, em analogia ao Método das Nuvens *hp*, a nuvem associada ao nó \mathbf{x}_j é definida pelos elementos que nele concorrem. No \mathbb{R}^1 , por exemplo, a PU é construída por funções lineares que vale 1 na posição do nó \mathbf{x}_j e zero nas posições dos nós adjacentes, como apresentado na Fig. 2.2.

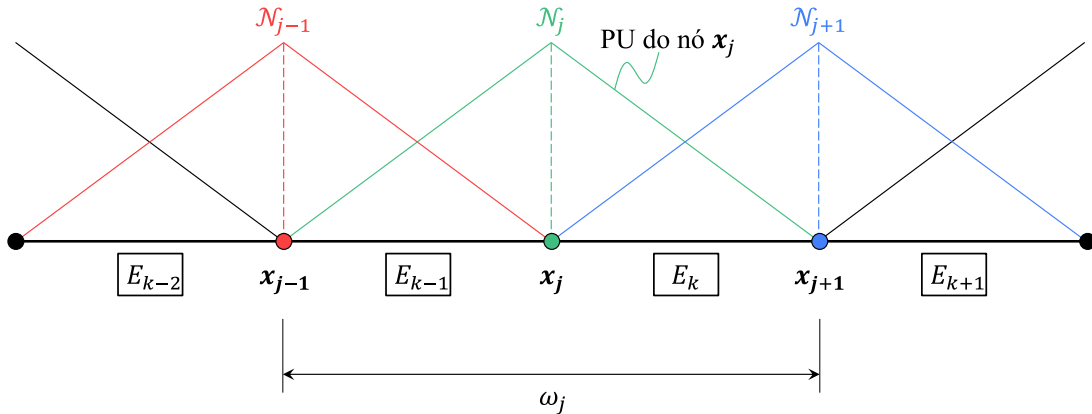


Figura 2.2: Partição da Unidade para elementos finitos em \mathbb{R}^1 (adaptado de Barros (2002)).

Na figura, \mathcal{N}_j denota a função utilizada para descrever a PU associada ao nó \mathbf{x}_j e, assim, tem-se:

$$\sum_{j=1}^n \mathcal{N}_j(\mathbf{x}) = 1 \quad \forall \mathbf{x} \in \bar{\Omega}, \quad (2.2)$$

relembrando que $\bar{\Omega} \subset \cup_{j=1}^n \{\omega_j\}$ é o domínio fechado do problema.

Uma vez que a PU está estabelecida, deve-se definir a função de aproximação local, também chamada de função de enriquecimento e aqui denotada por $L_{ji}(\mathbf{x})$. As funções de enriquecimento são selecionadas conforme o tipo de problema estudado (Alves, 2012). Tais funções têm origem diversa, como será abordado na Seção 2.2. Em geral, o que se busca é a seleção de funções que representem bem o comportamento esperado na região de interesse, isto é, a região de suporte da nuvem associada ao nó que receberá o enriquecimento. Portanto, é necessário um conhecimento *a priori* do comportamento da solução para que as funções de enriquecimento sejam definidas de forma adequada. Essa é uma limitação do MEFG, afinal, o desconhecimento do comportamento exato do problema é uma das razões para o uso de métodos numéricos aproximados.

No MEFG, o conjunto das q_j funções de forma $\{\varphi_{ji}\}_{i=1}^{q_j}$ associadas ao nó \mathbf{x}_j resulta do processo de enriquecimento da PU, ilustrado na Fig. 2.3. Para o nó \mathbf{x}_j , as funções de forma do MEFG são obtidas pelo produto entre a função básica $\mathcal{N}_j(\mathbf{x})$ pelas funções de enriquecimento, isto é,

$$\{\varphi_{ji}(\mathbf{x})\}_{i=1}^{q_j} = \mathcal{N}_j(\mathbf{x}) \times \{L_{ji}(\mathbf{x})\}_{i=1}^{q_j}. \quad (2.3)$$

Observa-se que a Eq. (2.3) não possui somatório em j , uma vez que se refere ao conjunto de funções de forma associada a um nó. Ainda, a Eq. (2.3) evidencia algumas características que conferem significativa flexibilidade ao MEFG:

1. Quando $L_{ji} = 1$, a função de forma obtida para o MEFG é idêntica à do MEF.
2. Diferentes conjuntos de funções L_{ji} podem ser definidos para diferentes nós (Oden, Duarte e Zienkiewicz, 1996). Como consequência, por exemplo, os nós localizados em uma região do domínio podem ser enriquecidos com funções senoidais, enquanto os demais, com polinômios.

Associando as informações apresentadas nos itens 1 e 2, estende-se que o MEFG permite que apenas parte dos nós seja enriquecida, enquanto os demais nós estarão associados somente às funções de forma tradicionais do MEF. Ainda que o processo de enriquecimento seja tão flexível, podendo variar entre os nós, há continuidade da solução aproximada nas interfaces entre os elementos (Duarte, Babuška e Oden, 2000). A Fig. 2.4 ilustra a flexibilidade na definição das funções de forma do MEFG.

No MEFG, a função aproximadora $\mathbf{u}(\mathbf{x})$ é da forma

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}^{MEF}(\mathbf{x}) + \mathbf{u}^{enr}(\mathbf{x}), \quad (2.4)$$

em que \mathbf{u}^{MEF} é a aproximação obtida pela formulação convencional do MEF, enquanto \mathbf{u}^{enr} corresponde à parcela da solução associada ao enriquecimento da PU. Naturalmente, a parcela \mathbf{u}^{MEF} depende somente das funções de forma do MEF, $\mathcal{N}_j(\mathbf{x})$, e dos respectivos deslocamentos nodais \mathbf{u}_j e é definida como

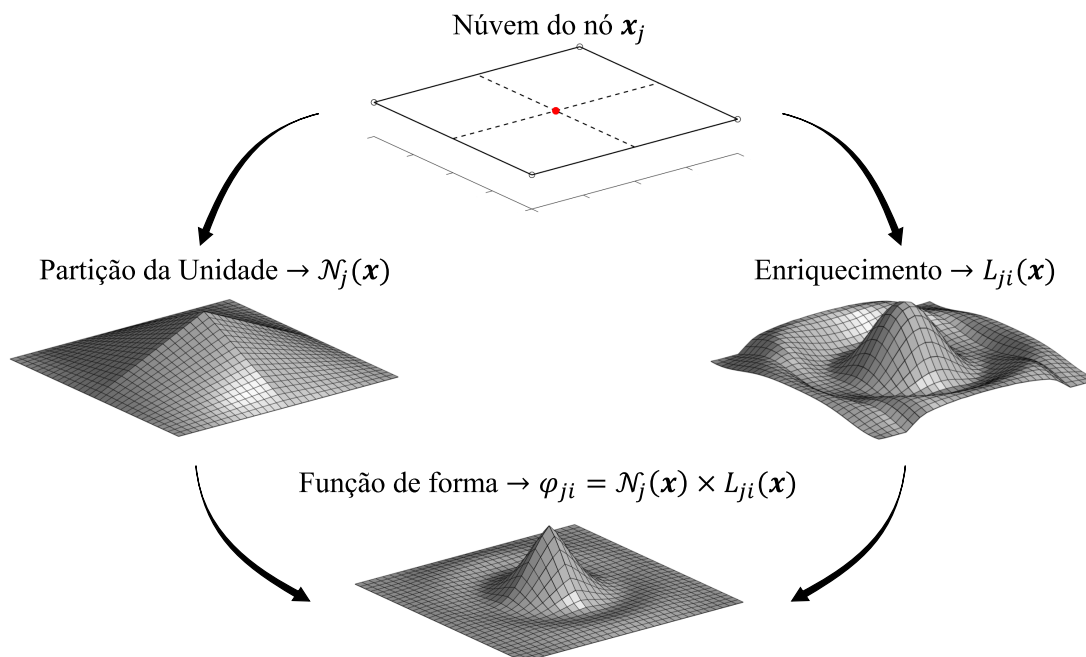


Figura 2.3: Processo de enriquecimento de uma PU.

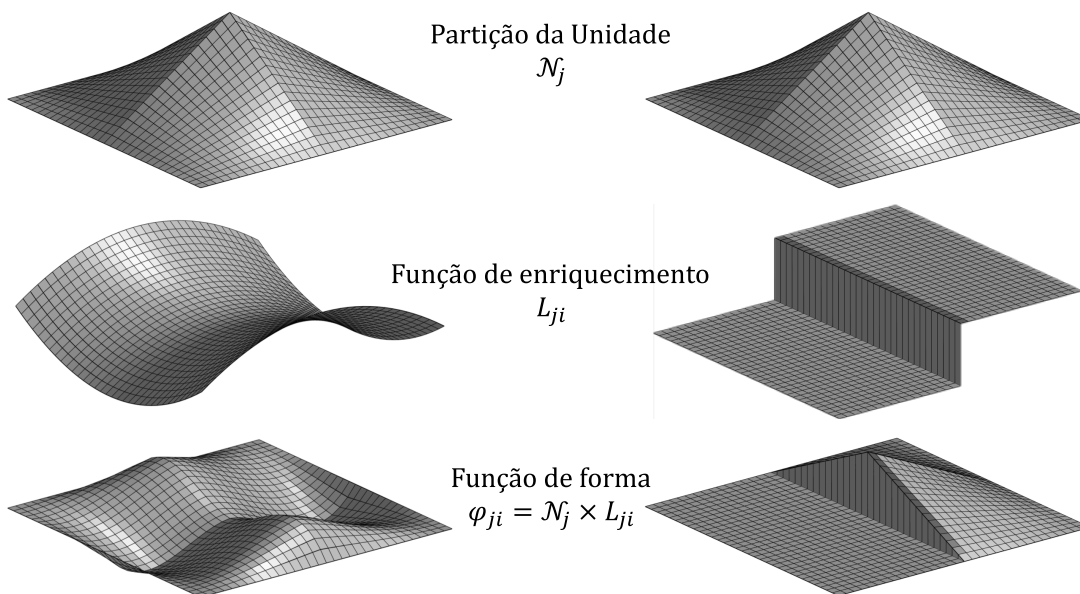


Figura 2.4: Flexibilidade nas funções de forma obtidas pelo MEFG.

$$\mathbf{u}^{MEF}(\mathbf{x}) = \sum_{j=1}^n \mathcal{N}_j(\mathbf{x}) \mathbf{u}_j. \quad (2.5)$$

Analogamente, a parcela associada ao enriquecimento depende das funções de forma do MEF, definidas na Eq. (2.3), e do parâmetro nodal associado, aqui denotado \mathbf{b}_{ji} . Assim, tem-se

$$\begin{aligned} u^{enr}(\mathbf{x}) &= \sum_{j=1}^n \varphi_j(\mathbf{x}) \mathbf{b}_{ji} \\ &= \sum_{j=1}^n \mathcal{N}_j(\mathbf{x}) \left(\sum_{i=2}^{q_j} L_{ji}(\mathbf{x}) \mathbf{b}_{ji} \right). \end{aligned} \quad (2.6)$$

Finalmente, substituindo as Eqs. (2.5) e (2.6) na Eq. (2.4), obtém-se a forma hierárquica da solução aproximada obtida pelo MEF como

$$\mathbf{u}(\mathbf{x}) = \underbrace{\sum_{j=1}^n \mathcal{N}_j(\mathbf{x}) \mathbf{u}_j}_{\mathbf{u}^{MEF}} + \underbrace{\sum_{j=1}^n \mathcal{N}_j(\mathbf{x}) \sum_{i=2}^{q_j} L_{ji}(\mathbf{x}) \mathbf{b}_{ji}}_{\mathbf{u}^{enr}}, \quad (2.7)$$

que pode ser rearranjada como

$$\mathbf{u}(\mathbf{x}) = \sum_{j=1}^n \mathcal{N}_j(\mathbf{x}) \left\{ \mathbf{u}_j + \sum_{i=2}^{q_j} L_{ji}(\mathbf{x}) \mathbf{b}_{ji} \right\}. \quad (2.8)$$

2.2 Método dos Elementos Finitos Generalizados com enriquecimento global-local

O MEF tem como característica o caráter genérico das funções de enriquecimento. Essas funções podem ser de qualquer natureza, inclusive provenientes de soluções numericamente obtidas. Duarte e Kim (2008) exploraram essa característica e propuseram o Método dos Elementos Finitos Generalizados com funções de enriquecimento global-local (MEFG^{gl}). O método pode ser entendido como uma abordagem combinada entre o MEF e a estratégia Global-Local do MEF, proposta por Noor (1986). Aqui, o problema é dividido em duas escalas. A escala global contém todo o domínio do problema, enquanto a escala local é um subdomínio da escala global e é definida nas regiões em que ocorrem fenômenos localizados de interesse. São exemplos de tais fenômenos: detalhes geométricos que alteram o campo de deformações, descontinuidades, singularidades, plasticidade e contatos.

Uma análise pelo MEF^{gl} se inicia pela solução do problema global. Essa solução é obtida com uma malha grosseira que não representa os fenômenos localizados. Chamada de solução inicial, é utilizada para a definição das condições de contorno do problema local. No problema local, a solução é procedida considerando todos os fenômenos localizados

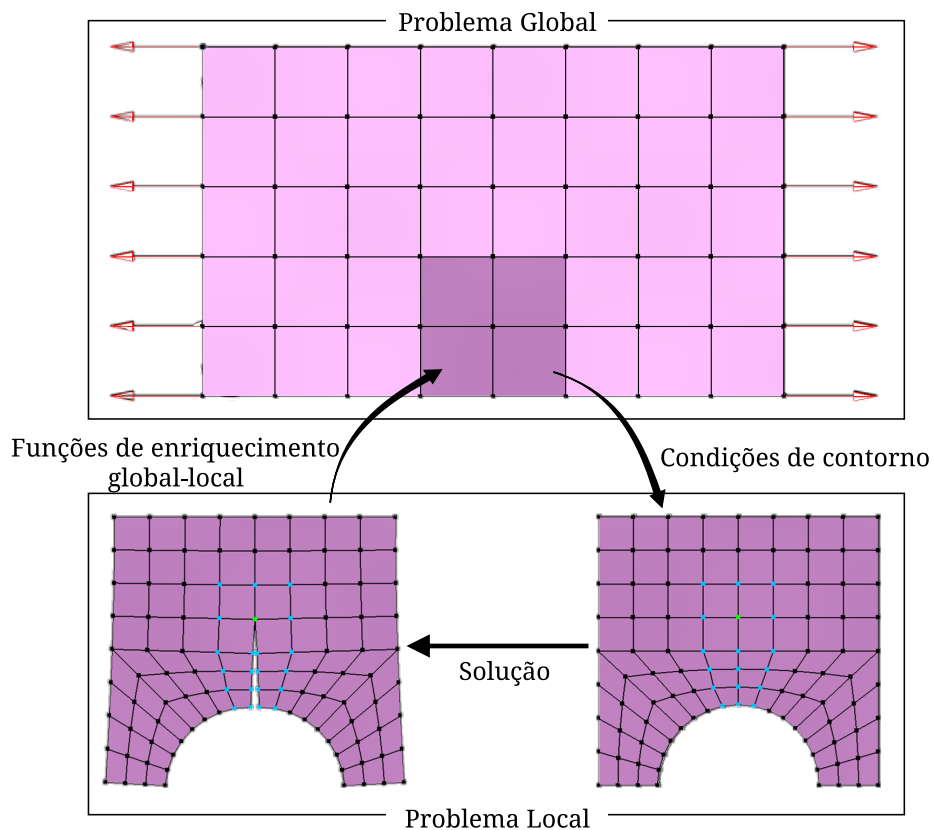


Figura 2.5: Procedimento de solução pelo $MEFG^{gl}$.

adequadamente e, em geral, a malha utilizada será refinada (Campos, 2020).

A solução numérica local é, então, utilizada para a obtenção de funções de enriquecimento global-local. O problema global é enriquecido com essas funções e uma nova solução global é obtida. Essa nova solução considera os efeitos localizados por meio do enriquecimento realizado e é denominada solução enriquecida.

O procedimento de solução está ilustrado na Fig. 2.5 e pode ser resumido nos seguintes passos:

Passo 1 *Solução global inicial*: modela-se o problema em escala global e se obtém a solução global inicial. Essa solução é, em geral, obtida pelo MEF utilizando malha grosseira.

Passo 2 *Solução local*: a solução da escala global é utilizada para a definição de condições de contorno do problema local. Então, a solução local é obtida. Aqui pode-se utilizar tanto o MEF quanto o MEFg.

Passo 3 *Enriquecimento global-local*: a solução local obtida numericamente no Passo 2 é utilizada para a construção de funções de enriquecimento global-local. O modelo global é enriquecido com essas funções.

Passo 4 *Solução global enriquecida*: o problema global é outra vez solucionado, agora por meio do MEFg. A solução global enriquecida considera os efeitos dos fenômenos

localizados pelas funções de enriquecimento global-local.

Passo 5 Ciclos global-local: se necessário, a solução global enriquecida pode ser utilizada no Passo 2, dando início a um novo ciclo de obtenção de funções de enriquecimento global-local.

Segundo Noor (1986), as principais limitações e dificuldades encontradas ao se aplicar a abordagem Global-Local no MEF estão associadas à interface entre as duas escalas. Kim, Pereira e Duarte (2009) indicam que tais limitações são superadas por meio do enriquecimento do problema global com as funções provenientes da solução local obtida numericamente.

A escala local pode abranger um ou mais problemas, associados a uma ou mais sub-regiões do domínio. Podem ser realizados ciclos de solução, isto é, ciclos MEFG^{gl} . Dessa forma, o MEFG^{gl} torna-se interessante por facilitar significativamente o processo de modelagem e de solução de problemas extensos que apresentam fenômenos localizados.

2.2.1 Formulação

A seguir, apresenta-se a formulação do MEFG^{gl} . Esta seção foi construída a partir dos trabalhos de Duarte e Kim (2008) e Alves (2012).

A solução inicial do problema global, \mathbf{u}_G^0 , está associada ao domínio global do problema, aqui denotado $\bar{\Omega}_G = \Omega_G + \partial\Omega_G$. O domínio local é, então, definido como um subdomínio do global, isto é, $\bar{\Omega}_L \subset \bar{\Omega}_G$. Os fenômenos locais ocorrem apenas em $\bar{\Omega}_L = \Omega_L + \partial\Omega_L$ e a solução desse domínio é denotada \mathbf{u}_L . A Fig. 2.6 ilustra o problema descrito.

O contorno do domínio local, $\partial\Omega_L$, é composto por três parcelas: $\partial\Omega_L^D = \partial\Omega_L \cap \partial\Omega_G^D$, que possui interseção com a parte do contorno do domínio global na qual foram definidas condições de contorno essenciais (contorno de Dirichlet); $\partial\Omega_L^N = \partial\Omega_L \cap \partial\Omega_G^N$, que possui

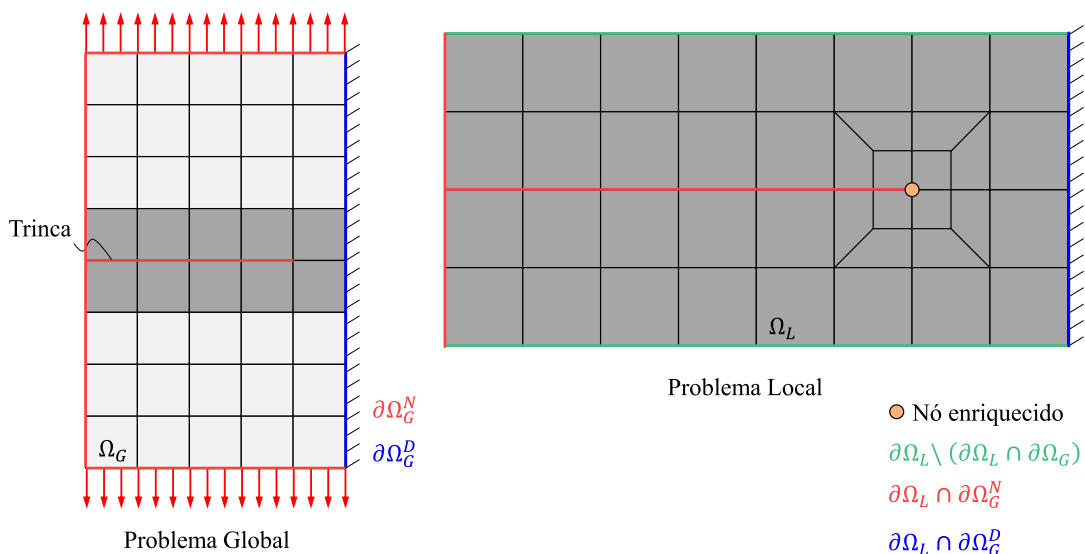


Figura 2.6: Domínios global e local no MEFG^{gl} . Adaptado de Alves (2012).

interseção com a parte do contorno do domínio global na qual foram definidas condições de contorno naturais (contorno de Neumann); e $\partial\Omega_L^C = \partial\Omega_L \setminus (\partial\Omega_L \cap \partial\Omega_G)$ é o contorno do domínio local complementar aos dois anteriores.

Na abordagem adotada nesse trabalho, o contorno da escala local não faz interseção com o contorno de Dirichlet do problema global. O problema associado consiste em encontrar $\mathbf{u}_L \in X_L(\Omega_L) \subset H^1(\Omega_L)$ tal que $\forall \mathbf{v}_L \in X_L(\Omega_L)$,

$$\begin{aligned} \int_{\Omega_L} \boldsymbol{\sigma}(\mathbf{u}_L) : \boldsymbol{\varepsilon}(\mathbf{v}_L) d\mathbf{x} + \kappa \int_{\partial\Omega_L^C} \mathbf{u}_L \cdot \mathbf{v}_L d\mathbf{s} = \\ \int_{\partial\Omega_L^N} \tilde{\mathbf{t}} \cdot \mathbf{v}_L d\mathbf{s} + \int_{\partial\Omega^C} (\mathbf{t}(\mathbf{u}_G^0) + \kappa \mathbf{u}_G^0) \cdot \mathbf{v}_L d\mathbf{s}, \end{aligned} \quad (2.9)$$

em que $X_L(\Omega_L)$ é o espaço de discretização do modelo local, que está contido em um espaço de Hilbert de ordem 1 definido no domínio local, $H^1(\Omega_L)$, e \mathbf{v}_L são as funções teste associadas ao espaço de solução do problema local. $\boldsymbol{\sigma}$ é o tensor de tensões de Cauchy, $\boldsymbol{\varepsilon}$ é o tensor de deformações e \mathbf{t} é o vetor de forças generalizadas. Os valores prescritos para os deslocamentos e para as tensões generalizadas são representados por $\tilde{\mathbf{u}}$ e $\tilde{\mathbf{t}}$, respectivamente.

A variável κ é o parâmetro de rigidez de Cauchy e está associada às condições de contorno do problema local. Esse tema não será explorado neste texto por fugir do seu escopo. No contexto do MEF gl , κ deve ser definido de acordo com a estratégia de transferência de condições de contorno entre Ω_G e Ω_L :

- Condições de contorno naturais (Neumann): adota-se $\kappa = 0$.
- Condições de contorno essenciais (Dirichlet): adota-se $\kappa \gg 1$.
- Condições de contorno mistas (Cauchy): κ assume um valor entre os dois anteriores. Para maiores detalhes, sugere-se como referência o trabalho de Kim, Pereira e Duarte (2009).

Na Eq. (2.9), o termo $\mathbf{t}(\mathbf{u}_G^0)$ refere-se às forças generalizadas computadas na escala global. Essa informação é transferida para o problema local por meio da relação de Cauchy e pela lei Hooke generalizada. Dessa forma, tem-se

$$\mathbf{t}(\mathbf{u}_G^0) = \hat{\mathbf{n}} \cdot \boldsymbol{\sigma}(\mathbf{u}_G^0) = \hat{\mathbf{n}} \cdot (\mathbf{C} : \boldsymbol{\varepsilon}(\mathbf{u}_G^0)), \quad (2.10)$$

em que $\hat{\mathbf{n}}$ é o vetor normal à $\partial\Omega_L^C$ e \mathbf{C} é o tensor de Hooke.

A solução aproximada $\mathbf{u}_L(\mathbf{x})$ pode, então, ser obtida numericamente. O objetivo, agora, é fazer com que o comportamento capturado por essa solução seja levado para a solução global. Nesse sentido, $\mathbf{u}_L(\mathbf{x})$ é utilizado para a construção de funções de forma do modelo global. Isso é realizado fazendo-se $L_{ji}(\mathbf{x}) = \mathbf{u}_L(\mathbf{x})$ na Eq. (2.3), isto é,

$$\varphi_j(\mathbf{x}) = \mathcal{N}_j(\mathbf{x}) \times \mathbf{u}_L(\mathbf{x}). \quad (2.11)$$

É importante lembrar que, na Eq. (2.11), \mathcal{N}_j são as funções de base utilizadas para a construção da PU do problema global – em geral, funções de forma convencionais do MEF.

Por fim, resta resolver o problema global enriquecido com a solução local. O carácter hierárquico do método manifesta-se na atualização do sistema de equações associado. Conforme Alves (2012), o sistema de equações associado ao problema global enriquecido é definido por

$$\begin{bmatrix} \mathbf{K}_G^0 & \mathbf{K}_G^{0,gl} \\ \mathbf{K}_G^{gl,0} & \mathbf{K}_G^{gl} \end{bmatrix} \begin{bmatrix} \mathbf{u}_G^0 \\ \mathbf{u}_G^{gl} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_G^0 \\ \mathbf{f}_G^{gl} \end{bmatrix}, \quad (2.12)$$

em que os índices gl estão associados aos graus de liberdade introduzidos pelo enriquecimento global-local, ao passo que os índices 0 referem-se aos graus de liberdade do problema global inicial.

2.2.2 Considerações sobre o MEF G^{gl}

A Eq. (2.11) estabelece as funções de forma do problema global enriquecido. O espaço de aproximação do problema global é, portanto, aumentado devido a introdução dos graus de liberdade associados ao enriquecimento. Essa atualização do problema global, contudo, não exige qualquer alteração na sua malha em termos do número de nós e de elementos. Isto é, são introduzidos graus de liberdade, mas apenas as funções de forma do problema global são alteradas.

Alves (2012) destaca que o enriquecimento apresentado na Eq. (2.11) não precisa ser o único enriquecimento utilizado no problema global. Se for conveniente, o enriquecimento global-local pode ser utilizado junto com outras funções de enriquecimento.

Duarte e Kim (2008) investigaram a influência da solução global inicial na qualidade do enriquecimento global-local obtido. Constataram que a qualidade da aproximação do problema global inicial possui influência na qualidade da solução final. Essa influência, contudo, não é significativa. O problema global pode ser solucionado por meio de *malha grosseira* sem que haja prejuízo considerável na solução final. Essas conclusões também foram observadas por Alves (2012). Neste contexto, o termo *malha grosseira* é utilizado em relação às características do problema localizado.

O trabalho de Duarte e Kim (2008) também buscou avaliar a influência da abrangência do domínio local na qualidade da solução obtida. Os resultados obtidos indicaram que, em geral, o domínio local deve se estender em torno da região com o comportamento distinto do global, ou seja, domínios locais extensos não são necessários. Trabalhos posteriores, contudo, indicam que esse comportamento depende do problema em estudo. Alves (2012), por exemplo, indica que a abrangência do domínio local pode influenciar na qualidade final da solução de problemas de mecânica da fratura.

Alves (2012) também avaliou se a quantidade de nós do modelo global que recebem o

enriquecimento influencia na solução final. Os resultados indicaram que o número de nós enriquecidos tem influência significativa na solução final. A solução melhora ao passo que mais nós recebem o enriquecimento. É importante, contudo, observar que quanto mais nós enriquecidos, maior o custo computacional da solução.

O'Hara, Duarte e Eason (2009) propuseram um processo iterativo como forma de reduzir o impacto negativo causado pela imposição de condições de contorno no problema local. A proposta foi avaliada para a solução de problemas de condução térmica. Nesse procedimento, a sequência de solução *global-local-global enriquecido* é repetida algumas vezes. Assim, permite-se que a aproximação global enriquecida possa aprimorar as condições de contorno impostas no problema local. Repete-se a referida sequência até que se observe a convergência dos resultados do problema global enriquecido. Isto é, trata-se dos ciclos $MEFG^{gl}$ mencionados no Passo 5 do procedimento de solução pelo $MEFG^{gl}$.

Gupta, Kim e Duarte (2012) avaliaram o procedimento iterativo de solução global-local para problemas da Mecânica da Fratura Linear Elástica, bem como a influência do que foi denominado *buffer zone*. Esse termo refere-se à região no problema global coincidente com o subdomínio local, que não recebe enriquecimento global-local. Observa-se que o incremento dessa região reduz os efeitos introduzidos pela imposição de condições de contorno.

O uso de ciclos $MEFG^{gl}$ e a *buffer zone* são estratégias para reduzir o erro associado a transferência de condições de contorno entre as escalas global e local. Não são todos os problemas que demandam o uso dessas estratégias. Quando necessário, contudo, o uso de uma dessas duas opções é suficiente para a obtenção de soluções adequadas. Vale ressaltar que a aplicação dessas estratégias oneram a solução do problema.

Em relação ao custo computacional da solução, Duarte e Kim (2008) destacaram que existem situações nas quais o $MEFG^{gl}$ pode ser mais eficiente do que as análises por meio do MEF e do MEEG. Segundo os autores, isso ocorre devido ao custo computacional para a solução de sistemas lineares aumentar em taxa maior do que a linear em relação ao tamanho do problema. Ou seja, em geral, é mais eficiente resolver diversos problemas locais menores do que um grande problema global. Além disso, a divisão de um problema em problemas menores reduz o consumo de memória, bem como facilita o uso de computação paralela. A submodelagem do problema também implica em conveniências de ordem prática, conforme comentado na seção seguinte. Ainda em relação ao desempenho computacional do $MEFG^{gl}$, Alves (2012) comenta que, a depender do problema avaliado, o método é eficiente, pois é capaz de se aproximar da solução de referência utilizando um número muito reduzido de graus de liberdade em comparação com o MEF.

2.3 Estratégias de acoplamento

Modelos simples são capazes de simular com facilidade e boa precisão diversos problemas de mecânica dos sólidos. Com frequência, contudo, ocorrem situações um pouco mais complexas que demandam maior cuidado durante a modelagem computacional. Em geral, as complexidades se restringem a certas regiões do problema, que podem ser estudadas por meio de modelos de menor abrangência, porém, mais precisos.

Nesse cenário, diversas pesquisas no âmbito dos métodos numéricos e da mecânica computacional se dedicaram à investigação de estratégias para solucionar problemas multiescala e/ou que envolvam a decomposição do domínio. Tem sido observado o desenvolvimento de uma grande variedade de métodos computacionais que, segundo Duval, Passieux, Salaün e Guinard (2014), podem ser classificados em dois grupos: os que buscam realizar o enriquecimento do modelo em elementos finitos e os que se destinam ao acoplamento entre modelos.

No primeiro grupo, o objetivo é enriquecer as funções de forma dos elementos finitos com funções específicas capazes de melhorar a aproximação originalmente definida (Barros, 2002). Podem ser citados os diversos métodos baseados na PU, com destaque para o MEFG, abordado na Seção 2.1, por ser bastante difundido. O MEFG^{gl}, mais recente e tema da Seção 2.2, também pertence a essa classe de métodos.

O segundo grupo consiste em métodos de solução para problemas multiescala utilizando mais de um modelo numérico. Nesses métodos, o objetivo é a obtenção de uma solução final que considere a influência entre as escalas (Guidault, Allix, Champaney e Cornuault, 2008). O domínio é investigado em uma escala aumentada, mais abrangente e menos detalhada, e uma solução \mathbf{s}_A é obtida. Fenômenos restritos à pequenas regiões do domínio são avaliados em escala reduzida, de menor abrangência e mais rica em detalhes, e sua solução \mathbf{s}_B é utilizada como correção da solução \mathbf{s}_A . A solução final \mathbf{s} é obtida por um procedimento que seja capaz de acoplar as soluções \mathbf{s}_A e \mathbf{s}_B . Isso deve ser realizado garantindo-se a continuidade de deslocamentos e o equilíbrio de forças nas interfaces entre as escalas (Ladevèze et al., 2001).

No uso prático, é comum que uma das escalas seja melhor avaliada por um *software* que não apresenta os recursos necessários ou desejados para a avaliação das demais escalas. Ainda que bastante robustos, poucos pacotes comerciais de análise pelo MEF, por exemplo, disponibilizam o MEFG. Podem existir programas desenvolvidos no ambiente de pesquisa, por outro lado, que disponibilizam os recursos necessários para esse tipo de análise (Malekan, 2017, Alves, 2012).

Outro cenário que motiva a submodelagem, nome muitas vezes dado ao processo de modelagem com decomposição de domínio (Gendre et al., 2011), ocorre quando um modelo computacional atinge extensão tal que sua manipulação via interface gráfica fica lenta e pouco responsiva. O usuário vê a necessidade de realizar a decomposição do problema em

mais de um modelo por razões de produtividade.

Há, ainda, outro fator prático que pode exigir a decomposição do domínio. Como comentado na Seção 2.2.2, o custo computacional para a solução de sistemas lineares cresce mais rapidamente do que o tamanho do problema. A depender do cenário, um dado *hardware* pode não ser capaz de atender à demanda computacional para a simulação de um problema utilizando apenas um grande e único modelo, mas pode ser capaz de realizar a mesma simulação utilizando vários modelos de menor extensão.

A demanda para o uso da submodelagem, somada ao desenvolvimento de tecnologias de computação paralela e de multiprocessamento, coloca em relevância os métodos de acoplamento (Duval et al., 2014). Existem métodos de acoplamento em abundância e a maioria deles utiliza procedimentos iterativos e/ou recursivos, muitas vezes combinados com estratégias de manipulação matricial. Podem ser citados diversos trabalhos nessa linha, como Przemieniecki (1963), Bjørstad e Widlund (1986), Wyart et al. (2008) e Gendre, Allix e Gosselet (2011). Em todas essas abordagens, para que o acoplamento seja realizado, é necessário realizar alterações nas matrizes de rigidez e/ou nas estratégias de solução (*solvers*) envolvidas. Essas características conferem a esses métodos a denominação de *acoplamento intrusivo*.

A principal desvantagem dos acoplamentos intrusivos reside no fato de que nem sempre é possível realizar as alterações necessárias para sua implementação. Salvo poucas exceções, o código-fonte dos programas computacionais de análise pelo MEF voltados à indústria é fechado. Não é possível, assim, que terceiros possam consultar ou modificar tais códigos. Essa característica reduz o potencial de aplicação dos métodos de acoplamento intrusivo.

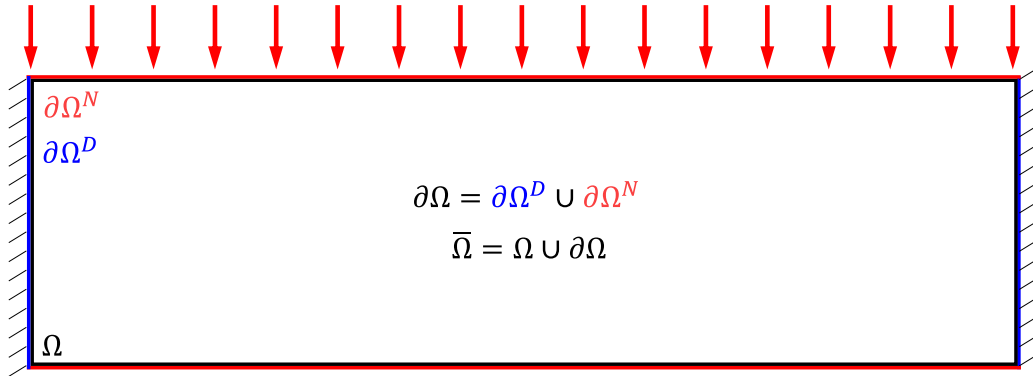
Naturalmente, diversas pesquisas foram realizadas com o objetivo de desenvolver métodos de *acoplamento não intrusivo*. Nesse tipo de acoplamento, a implementação dispensa a alteração dos programas computacionais ou das estratégias de solução (*solvers*) envolvidas (Duval et al., 2014). Devido à essa característica, o acoplamento não intrusivo permite a associação de diversos programas, sejam eles de código aberto ou não, para a solução de uma vasta gama de problemas.

Na última década, o acoplamento não intrusivo foi utilizado em conjunto com o MEEG para a análise de problemas envolvendo propagação de trincas (Gupta, Pereira, Kim, Duarte e Eason, 2012) e plasticidade (Gendre, Allix e Gosselet, 2011). Guguin, Allix, Gosselet e Guinard (2014) utilizaram o acoplamento não intrusivo para a solução de problemas multidimensionais (2D-3D) pelo MEF, bem como Guinard, Bouclier, Toniolli e Passieux (2018) utilizaram esse tipo de acoplamento para a solução de problemas complexos e de grande escala por meio da abordagem global-local aplicada ao MEF.

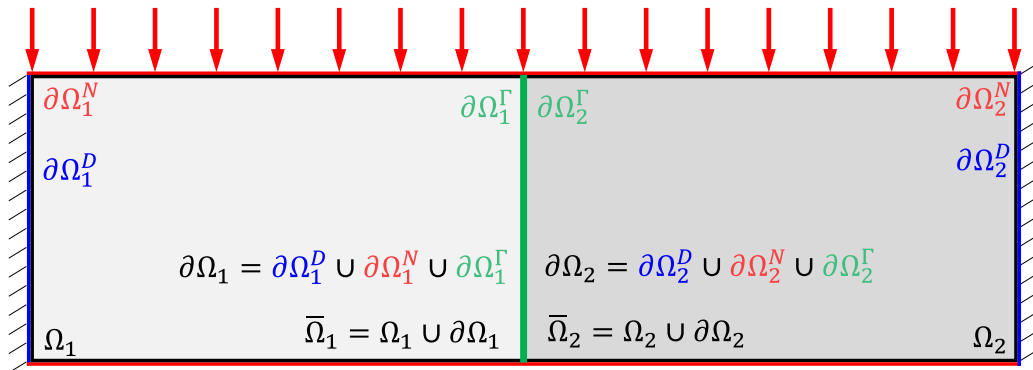
2.3.1 Problema de decomposição do domínio

Apresenta-se nesta seção o problema de decomposição de domínio. Após a decomposição, o acoplamento entre as soluções será proposto de forma não intrusiva. Esta seção é baseada nos trabalhos de Duval, Passieux, Salaün e Guinard (2014).

Seja considerado o problema apresentado na Fig. 2.7a, cujo domínio é $\bar{\Omega} = \Omega \cup \partial\Omega$. Sejam definidos os deslocamentos prescritos \mathbf{u}_D , impostos ao longo do contorno $\partial\Omega^D$, e as forças generalizadas \mathbf{f}_N , em $\partial\Omega^N$. Também define-se que $\partial\Omega = \partial\Omega^D \cup \partial\Omega^N$ e que $\partial\Omega^D \cap \partial\Omega^N = \emptyset$.



(a) Problema inicial.



(b) Decomposição do problema inicial.

Figura 2.7: Problema de decomposição de domínio.

O equilíbrio do problema é satisfeito pela equação

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{f}_B = \mathbf{0} \quad \text{em } \bar{\Omega}, \quad (2.13)$$

em que \mathbf{f}_B são as forças de corpo.

Sejam considerados o tensor de Hooke \mathbf{C} e o tensor de deformação $\boldsymbol{\varepsilon}$. Se X é o espaço de discretização, contido em um espaço de Hilbert de ordem 1 definido em Ω , $H^1(\Omega)$, a formulação fraca do problema consiste em encontrar $\mathbf{u} \in X(\Omega) \subset H^1(\Omega)$ tal

que $\forall \mathbf{v} \in X(\Omega)$,

$$\int_{\Omega} (\mathbf{C} : \boldsymbol{\varepsilon}(\mathbf{u})) : \boldsymbol{\varepsilon}(\mathbf{v}) \, d\mathbf{x} = \int_{\Omega} \boldsymbol{\sigma}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v}) \, d\mathbf{x} = \int_{\partial\Omega^N} \mathbf{t} \cdot \mathbf{v} \, d\mathbf{x} + \int_{\partial\Omega} \mathbf{f}_B \cdot \mathbf{v} \, d\mathbf{x}, \quad (2.14)$$

em que \mathbf{v} são as funções teste associadas ao espaço de solução do problema.

Seja, então, o domínio $\bar{\Omega}$ decomposto nos subdomínios $\bar{\Omega}_1 = \Omega_1 \cup \partial\Omega_1$ e $\bar{\Omega}_2 = \Omega_2 \cup \partial\Omega_2$, conforme apresentado na Fig. 2.7b. Os subdomínios $\bar{\Omega}_1$ e $\bar{\Omega}_2$ não se sobrepõem. Ainda assim, há uma região compartilhada entre eles, denominada interface entre $\bar{\Omega}_1$ e $\bar{\Omega}_2$ e definida como $\Gamma = \partial\Omega_1 \cap \partial\Omega_2$.

As condições de contorno do problema também são decompostas de acordo com os subdomínios. No domínio $\bar{\Omega}_1$, tem-se os deslocamentos prescritos $\mathbf{u}_{D,1}$ associados a $\partial\Omega_1^D$ e as forças generalizadas \mathbf{t}_1 que atuam em $\partial\Omega_1^N$. Para o $\bar{\Omega}_2$, em analogia, tem-se $\mathbf{u}_{D,2}$ e \mathbf{t}_2 associados aos contornos $\partial\Omega_2^D$ e $\partial\Omega_2^N$, respectivamente. do mesmo modo, a decomposição do domínio repercute no espaço das discretizações, de modo que

$$X_1 \subset H^1(\Omega_1) \quad \text{e} \quad X_2 \subset H^1(\Omega_2). \quad (2.15)$$

É preciso, então, definir como serão satisfeitas as condições de compatibilidade de deslocamentos e de equilíbrio de forças na interface Γ . Existem diversos métodos para garantir que essas condições sejam satisfeitas. O *mortar element method*, apresentado nos trabalhos de Bernardi, Maday e Patera (1989 e 1993), é capaz de impor as condições mencionadas sem que seja exigida a conformidade entre as discretizações X_1 e X_2 na interface Γ . Gander e Japhet (2013) propuseram, ainda, um algoritmo para a implementação computacional do método em espaços bi e tridimensionais.

Define-se, então, $X_\psi \subset L^2(\Gamma)$ como o espaço dos multiplicadores de Lagrange associados à interface. A imposição da compatibilidade de deslocamentos e do equilíbrio de forças pelo *mortar element method* conduz à seguinte formulação para a decomposição de Ω em Ω_1 e Ω_2 :

$$\begin{aligned} \mathbf{u}_1 &\in X_1, & \mathbf{u}_2 &\in X_2, & \boldsymbol{\psi} &\in X_\psi, \\ \forall \mathbf{v}_1 &\in X_1, & a_1(\mathbf{u}_1, \mathbf{v}_1) + b(\boldsymbol{\psi}, \mathbf{v}_1) &= l_1(\mathbf{v}_1) \\ \forall \mathbf{v}_2 &\in X_2, & a_2(\mathbf{u}_2, \mathbf{v}_2) - b(\boldsymbol{\psi}, \mathbf{v}_2) &= l_2(\mathbf{v}_2) \\ \forall \mathbf{p} &\in X_\psi, & b(\mathbf{p}, \mathbf{u}_1 - \mathbf{u}_2) &= 0, \end{aligned} \quad (2.16)$$

em que \mathbf{p} é a função teste associadas ao espaço dos multiplicadores de Lagrange $\boldsymbol{\psi}$. As formas bilineares a_1 e a_2 são associadas a $\bar{\Omega}_1$ e $\bar{\Omega}_2$, respectivamente, enquanto b é a forma bilinear que realiza o acoplamento em Γ . Os termos l_1 e l_2 são as formas lineares associadas às condições de contorno de Neumann e às forças de corpo. A seguir, todos esses termos

são definidos:

$$a_1(\mathbf{u}_1, \mathbf{v}_1) = \int_{\Omega_1} \boldsymbol{\sigma}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v}) \, d\mathbf{x} \quad (2.17)$$

$$a_2(\mathbf{u}_2, \mathbf{v}_2) = \int_{\Omega_2} \boldsymbol{\sigma}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v}) \, d\mathbf{x} \quad (2.18)$$

$$l_1(\mathbf{v}_1) = \int_{\Gamma_1^N} \mathbf{t} \cdot \mathbf{v}_1 \, d\mathbf{x} + \int_{\Gamma_1^N} \mathbf{f}_B \cdot \mathbf{v}_1 \, d\mathbf{x} \quad (2.19)$$

$$l_2(\mathbf{v}_2) = \int_{\Gamma_2^N} \mathbf{t} \cdot \mathbf{v}_2 \, d\mathbf{x} + \int_{\Gamma_2^N} \mathbf{f}_B \cdot \mathbf{v}_2 \, d\mathbf{x} \quad (2.20)$$

$$b(\mathbf{p}, \mathbf{u}_1 - \mathbf{u}_2) = \int_{\Gamma} \mathbf{p} \cdot (\mathbf{u}_1 - \mathbf{u}_2) \, d\mathbf{x}, \quad (2.21)$$

em que $\Gamma_1^N = \Gamma \cap \Omega_1^N$ e $\Gamma_2^N = \Gamma \cap \Omega_2^N$.

Cada domínio é discretizado por uma malha. Para o domínio Ω_1 , a malha é M_1 e ela possui n_1 graus de liberdade. A região de M_1 na interface Γ é denotada M_{Γ_1} e o número de graus de liberdade associado é n_{Γ_1} . Em analogia, M_2 , M_{Γ_2} , n_2 e n_{Γ_2} são as respectivas definições para o domínio Ω_2 . Por fim, a malha associada aos multiplicadores de Lagrange é $M_{\Gamma\psi}$ e $n_{\Gamma\psi}$ denota o número de graus de liberdade a eles associados.

No contexto do MEF, define-se os conjuntos de funções de forma φ_1 , φ_2 e φ_ψ associadas aos espaços X_1 , X_2 e X_ψ , respectivamente. Pode-se, então, definir a equação matricial do problema do acoplamento não intrusivo em termos das matrizes de rigidez, dos vetores de forças e das matrizes de acoplamento. As matrizes de rigidez associadas à Ω_1 e Ω_2 são obtidas pela integração das Eqs. (2.17) e (2.18), respectivamente, isto é

$$(K_1)_{ij} = \int_{\Omega_1} (C \cdot \boldsymbol{\varepsilon}(\varphi_1^i(\mathbf{x}))) : \boldsymbol{\varepsilon}(\varphi_1^j(\mathbf{x})) \, d\mathbf{x} \quad (2.22)$$

$$(K_2)_{ij} = \int_{\Omega_2} (C \cdot \boldsymbol{\varepsilon}(\varphi_2^i(\mathbf{x}))) : \boldsymbol{\varepsilon}(\varphi_2^j(\mathbf{x})) \, d\mathbf{x}. \quad (2.23)$$

Os vetores de força associados a cada um dos domínios são obtidos pelo uso das funções de forma nas Eqs. (2.19) e (2.20),

$$(f_1)_j = \int_{\Gamma_1^N} \mathbf{t} \cdot \varphi_1^j(\mathbf{x}) \, d\mathbf{x} + \int_{\Gamma_1^N} \mathbf{f}_B \cdot \varphi_1^j(\mathbf{x}) \, d\mathbf{x} \quad (2.24)$$

$$(f_2)_j = \int_{\Gamma_2^N} \mathbf{t} \cdot \varphi_2^j(\mathbf{x}) \, d\mathbf{x} + \int_{\Gamma_2^N} \mathbf{f}_B \cdot \varphi_2^j(\mathbf{x}) \, d\mathbf{x}, \quad (2.25)$$

e, finalmente, a aplicação da Eq. (2.21) nos domínios envolvidos resulta nas matrizes de acoplamento

$$(A_1)_{ij} = \int_{\Gamma} \varphi_\psi^i \cdot \varphi_1^j(\mathbf{x}) \, d\mathbf{x} \quad (2.26)$$

$$(A_2)_{ij} = \int_{\Gamma} \varphi_\psi^i \cdot \varphi_2^j(\mathbf{x}) \, d\mathbf{x}. \quad (2.27)$$

2.3.2 Acoplamento na estratégia global-local

Seja, novamente, o domínio $\bar{\Omega} = \Omega \cup \partial\Omega$ apresentado na seção anterior. Agora, sua decomposição é feita nos subdomínios local $\bar{\Omega}_L = \Omega_L \cup \partial\Omega_L$ e complementar $\bar{\Omega}_C = \Omega_C \cup \partial\Omega_C$. Essa decomposição é representada na Fig. 2.8 e não há sobreposição entre $\bar{\Omega}_L$ e $\bar{\Omega}_C$. Todo o desenvolvimento realizado na Seção 2.3.1 pode ser reproduzido para os subdomínios local e complementar e os resultados serão os mesmos, bastando apenas substituir os sub-índices 1 e 2 por C e L . Esse desenvolvimento pode ser encontrado em Duval, Passieux, Salaün e Guinard (2014).

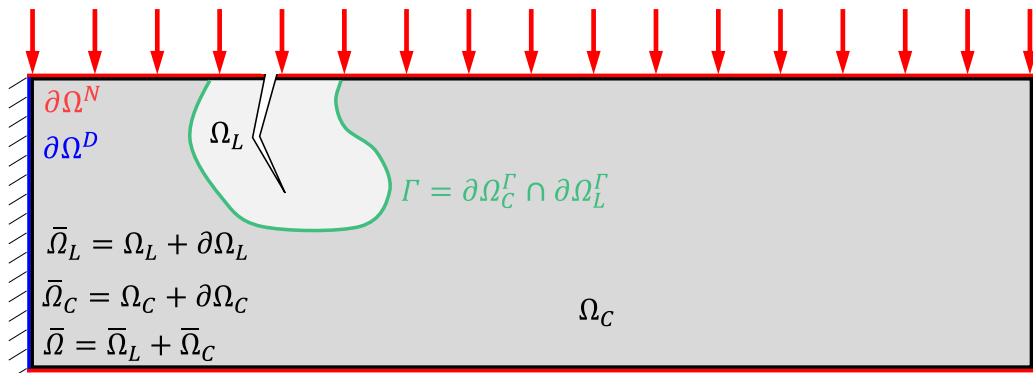
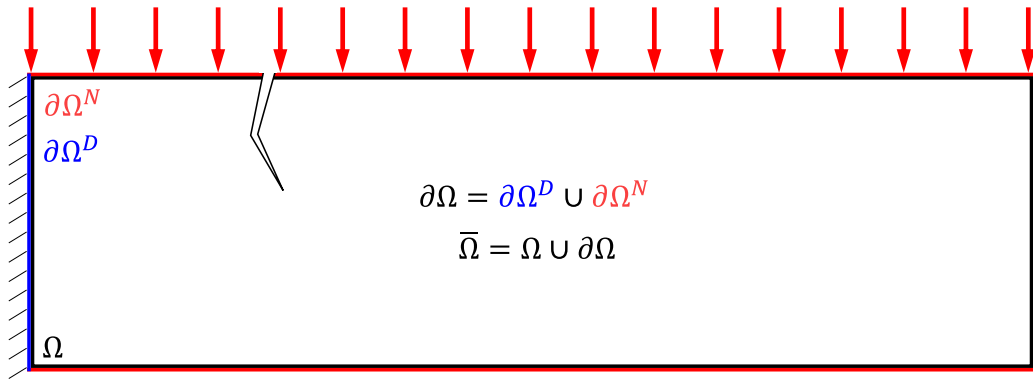


Figura 2.8: Problema global-local.

A seleção de funções de base para a representação do espaço X_ψ , associada à interface, não é simples. A aplicação de funções inapropriadas na interface pode causar inconsistências dos resultados obtidos (Wohlmuth, 2000). Duval, Passieux, Salaün e Guinard (2014), dentro do contexto da análise global-local, propuseram que as funções de forma convencionais do MEF definidas na interface do domínio local sejam utilizadas também como base dos multiplicadores de Lagrange. Tem-se, portanto, que a discretização da interface coincide com a discretização do domínio local na interface, isto é $M_{\Gamma\psi} = M_{\Gamma L}$ e $\varphi_\Gamma = \varphi_L$. Ainda segundo Duval, Passieux, Salaün e Guinard (2014), essa configuração foi testada em vários casos e nenhum problema foi encontrado ao se realizar a projeção entre as matrizes envolvidas.

Apenas para demonstração da decomposição realizada, o sistema de equações que *seria* solucionado no âmbito do MEF em sua formulação convencional é

$$\begin{bmatrix} \mathbf{K}_C & \mathbf{0} & \underline{\mathbf{A}}_C^\top \\ \mathbf{0} & \mathbf{K}_L & -\underline{\mathbf{A}}_L^\top \\ \underline{\mathbf{A}}_C^\top & -\underline{\mathbf{A}}_L^\top & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u}_C \\ \mathbf{u}_L \\ \boldsymbol{\psi} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_C \\ \mathbf{f}_L \\ \mathbf{0} \end{bmatrix}. \quad (2.28)$$

Com o objetivo de simplificar a notação utilizada, o símbolo $\underline{}$ foi utilizado para denotar o operador prolongamento de uma matriz, de modo que haja consistência nas dimensões dos elementos envolvidos nas operações matriciais. Por exemplo, $\underline{\mathbf{A}}_C$ é o prolongamento da matriz \mathbf{A}_C do espaço definido em $M_{\Gamma\psi} \times M_{\Gamma C}$ para $M_{\Gamma\psi} \times M_C$. Portanto, enquanto a matriz \mathbf{A}_C possui dimensões $n_{\Gamma\psi} \times n_{\Gamma C}$, $\underline{\mathbf{A}}_C$ é da ordem $n_{\Gamma\psi} \times n_C$, preenchendo com zeros os elementos adicionados devido a prolongação.

A Eq. (2.28) estabelece o acoplamento entre os domínios $\bar{\Omega}_C$ e $\bar{\Omega}_L$ e sua solução consiste em um *fixed point solver* (Allix e Gosselet, 2020). O objetivo é dissociar os domínios $\bar{\Omega}_C$ e $\bar{\Omega}_L$ durante a solução do problema. Nesse sentido, são adotadas duas escalas de solução. A escala local refere-se à $\bar{\Omega}_L$. A escala global, por sua vez, refere-se a todo o domínio do problema, isto é, $\bar{\Omega} = \bar{\Omega}_L \cup \bar{\Omega}_C$. O princípio básico consiste em calcular a diferença entre a reação dos modelos global e local na interface Γ a cada iteração e utilizar o resíduo para atualizar e solucionar novamente o problema global, que fornece as condições de contorno para o problema local.

Ainda que o acoplamento global-local apresentado nesta seção seja bastante difundido e robusto, existem limitações relevantes. Duval, Passieux, Salaün e Guinard (2014) destacam, nesse sentido, que:

1. O algoritmo é intrusivo.
2. Sua implementação não é trivial. Essa característica é mais evidente quando o domínio local é alterado durante a análise, como ocorre, por exemplo, em estudos de propagação de trincas.
3. Sua aplicação requer a montagem de duas matrizes de rigidez a cada iteração.
4. Pode ser necessário atualizar a malha do modelo global em função dos resultados do modelo local.

Além disso, tanto o tempo de modelagem quanto o tempo de solução aumentam quando existem muitos domínios locais. O uso do acoplamento não intrusivo, discutido na próxima seção, permite que tais limitações sejam superadas.

2.3.3 Acoplamento não intrusivo para análise global-local iterativa

Seja o problema apresentado na Fig. 2.9. O domínio global do problema é $\bar{\Omega} = \bar{\Omega}_C \cup \bar{\Omega}_L$, em que $\bar{\Omega}_L$ é o domínio local e $\bar{\Omega}_C$ é o seu domínio complementar em relação ao domínio global. O problema é novamente considerado em duas escalas. A escala global considera todo o domínio do problema, isto é, $\bar{\Omega}$. A escala local, por sua vez, é restrita ao domínio $\bar{\Omega}_L$ – ressaltando-se que podem existir mais de um problema local. A região $\Gamma = \Omega_C \cap \Omega_L$ é a interface.

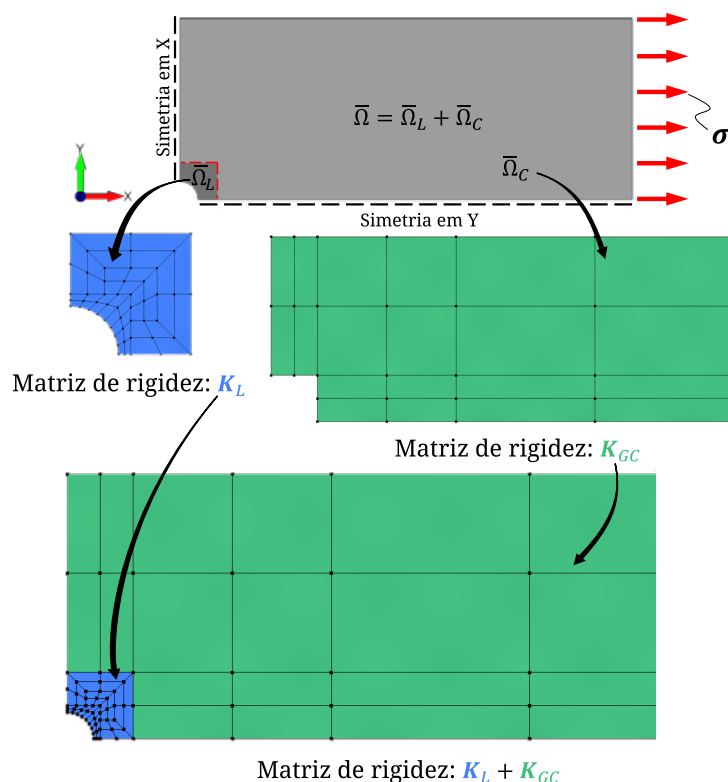


Figura 2.9: Solução de referência obtida pelo acoplamento não intrusivo para análise global-local.

A ideia central consiste em utilizar o modelo global para obter uma solução inicial que será utilizada como condição de contorno para o modelo local e, a partir da solução local, dar início a um processo iterativo. Na hipótese de convergência, esse processo se aproximará de uma solução de referência. Conforme apresentado na Fig. 2.9, essa solução corresponde àquela que teria sido obtida por um modelo no qual a discretização do modelo global é utilizada em $\bar{\Omega}_C$ e a discretização do modelo local, em $\bar{\Omega}_L$.

Sejam, então, as malhas M_{GC} e M_{GL} construídas para representar $\bar{\Omega}_C$ e $\bar{\Omega}_L$ no modelo global, respectivamente. A malha M_L refere-se ao modelo local – que abrange exclusivamente $\bar{\Omega}_L$, portanto.

Apresenta-se, a seguir, as grandezas¹ que compõem o sistema de equações associado

¹Nas equações a seguir, observar o uso do símbolo $_$ sob alguns termos. Consultar a discussão da

ao problema. No modelo global, a matriz de rigidez é $\mathbf{K} = \mathbf{K}_{GC} \cup \mathbf{K}_{GL}$ e o vetor de forças é $\mathbf{f} = \mathbf{f}_{GC} \cup \mathbf{f}_{GL}$. Por sua vez, no modelo local, a matriz de rigidez é \mathbf{K}_L e o vetor de forças é \mathbf{f}_L .

O acoplamento não intrusivo busca a obtenção da solução

$$\mathbf{u} = \mathbf{u}_C \in X_C \cup \mathbf{u}_L \in X_L, \quad (2.29)$$

de modo que a convergência da solução inicial \mathbf{u}^0 para a solução \mathbf{u} seja realizada sem que a matriz de rigidez \mathbf{K} seja alterada (Duval, Passieux, Salaün e Guinard, 2014). Na Eq. (2.29), X_C e X_L são os espaços de discretização de Ω_C e Ω_L , enquanto \mathbf{u}_C e \mathbf{u}_L são as soluções dos domínios complementar e local, respectivamente. A solução final \mathbf{u} deve ser tal que sejam satisfeitas as condições de compatibilidade de deslocamento e de equilíbrio de forças em Γ .

Lista-se, então, as seguintes vantagens associadas ao acoplamento não intrusivo no contexto da análise global-local, que serão comentadas na sequência:

1. Permite que o modelo global permaneça inalterado durante todo o processo de obtenção da solução.
2. Permite que o modelo global e o modelo local sejam solucionados por códigos distintos. Caso conveniente, pode-se até utilizar metodologias distintas para a solução de cada um dos modelos.
3. Permite a redução do custo computacional para obtenção da solução final.

No contexto da análise global-local, há grande vantagem em se obter a solução final sem que o modelo global seja modificado (Tirvaudey, Chamoin, Bouclier e Passieux, 2020). Nesse cenário, a matriz de rigidez do modelo global pode ser montada e invertida apenas uma vez. As respostas obtidas pelos problemas locais não influenciam no cálculo dessa matriz de rigidez em nenhuma etapa da solução. Desse modo, há significativa redução de custos de computação em relação aos algoritmos intrusivos.

O fato de o problema local não influenciar na matriz de rigidez global também é a razão de ser possível o uso de códigos diferentes para resolver cada um dos modelos envolvidos (Tirvaudey et al., 2020). Por exemplo, a escala global pode ser resolvida por um programa que implementa o MEF, enquanto o problema local pode ser solucionado pelo MEEG. Além disso, o procedimento pode ser feito sem que um programa tenha que atualizar a sua matriz de rigidez em função dos resultados do outro. Por isso, não é necessário acesso ao código-fonte dos programas envolvidos para que o acoplamento seja realizado.

página 42 acerca da notação adotada. Ainda, o símbolo \cup foi utilizado em referência à soma matricial considerando a projeção dos elementos envolvidos para o mesmo espaço.

Segundo Duval, Passieux, Salaün e Guinard (2014), a principal limitação do acoplamento não intrusivo está na sua lenta convergência. Quanto maior a *diferença de rigidez*² na discretização de $\bar{\Omega}_L$ no modelo local e no modelo global, mais lenta é a convergência dos resultados na interface, conforme será explorado em detalhes na Seção 3.2.2. Ainda que esse fenômeno não tenha se mostrado relevante para problemas com efeitos não-lineares muito localizados (Gendre et al., 2009, Liu et al., 2014), tal característica pode, a princípio, limitar a aplicação do acoplamento não intrusivo no contexto da análise global-local. Nesse sentido, os trabalhos de Duval et al. (2014) e Gendre et al. (2009) buscaram estudar técnicas com o objetivo de promover a redução do número de iterações necessárias para a convergência da solução.

A classe de métodos que atualmente é referida como acoplamentos não intrusivos tem origem no trabalho publicado por Whitcomb em 1991. O autor propôs uma metodologia altamente não intrusiva para o acoplamento de soluções pelo MEF. Denominado *iterative global-local finite element analysis* (IGL-FEM ou MEF-GLI, em português), o método foi bastante aplicado no âmbito do MEF. Mais recentemente, o algoritmo Global-Local Iterativo (GLI) têm sido adaptado para ser implementado em outros métodos além do MEF. Podem ser citados os trabalhos de Duval, Passieux, Salaün e Guinard (2014), Allix e Gosselet (2020) e Li, O'Hara e Duarte (2021).

Nesse contexto, a próxima seção apresenta o MEF-GLI. Em seguida, a Seção 2.5.1 apresenta o MEF^{gl}-GLI, que consiste na associação do GLI com o Método dos Elementos Finitos Generalizados com enriquecimento global-local (MEFG^{gl}).

2.4 Método dos Elementos Finitos Global-Local Iterativo

Seja, agora, o problema apresentado na Fig. 2.10. O domínio local $\bar{\Omega}_L$ é definido pela região próxima ao furo, onde se espera perturbação no campo de deformações devido ao detalhe geométrico. O modelo local consiste em uma discretização refinada de $\bar{\Omega}_L$ e sua matriz de rigidez é \mathbf{K}_L . Na Fig. 2.10, as grandezas associadas ao modelo local estão em azul.

O modelo global discretiza grosseiramente todo o domínio $\bar{\Omega} = \bar{\Omega}_C \cup \bar{\Omega}_L$. A região do modelo global referente à $\bar{\Omega}_L$ tem \mathbf{K}_{GL} como sua matriz de rigidez. Da mesma forma, a região do modelo global referente ao domínio complementar está associada à matriz rigidez \mathbf{K}_{GC} . Essas parcelas constituem a rigidez do modelo global. Na Fig. 2.10, as informações associadas à $\bar{\Omega}_L$ estão em vermelho, enquanto as informações em verde se refere à $\bar{\Omega}_C$.

²As matrizes \mathbf{K}_{GL} e \mathbf{K}_L , em geral, possuem dimensões distintas. O termo diferença de rigidez (ou *stiffness gap*, termo utilizado na literatura estrangeira) refere-se à discretização do domínio $\bar{\Omega}_L$. No MEF, uma malha mais grosseira é mais rígida.

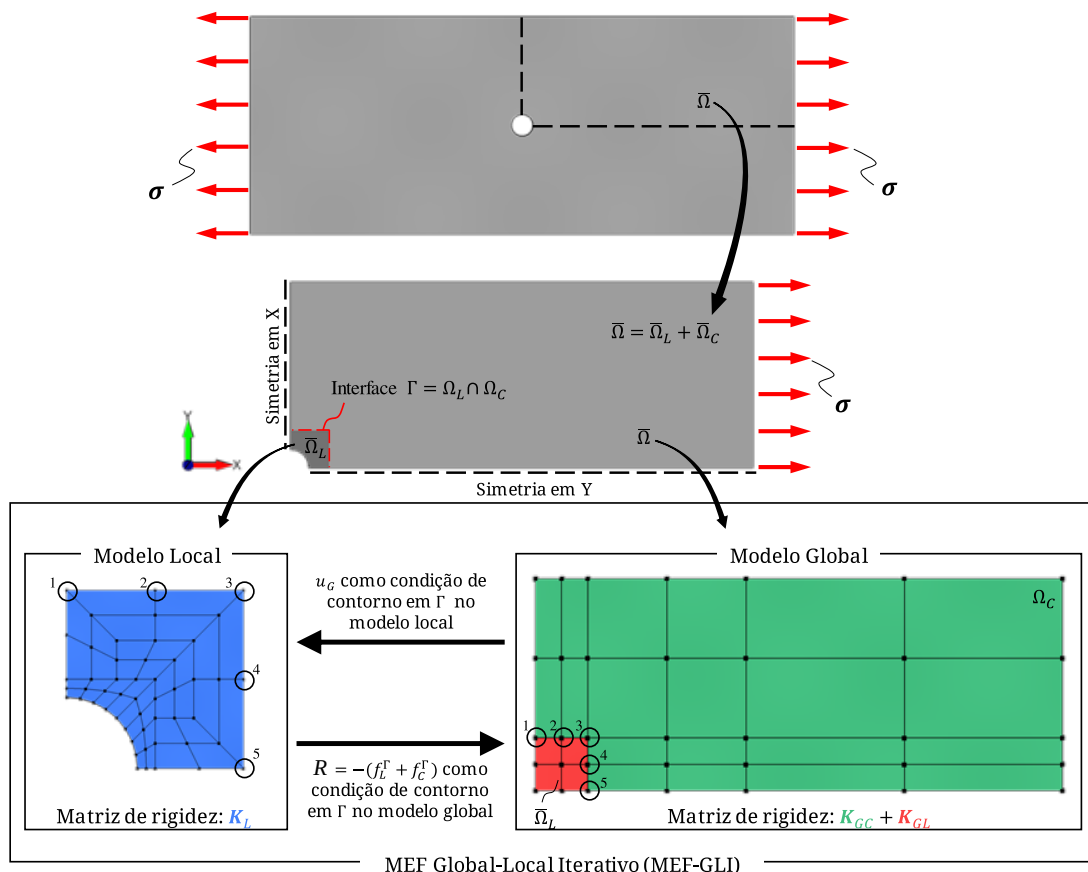


Figura 2.10: Processo de solução pelo MEF-GLI.

A seguir, são apresentadas algumas equações matriciais nas quais as notações de prolongamento das matrizes e dos vetores envolvidos serão omitidas com o objetivo de simplificar as equações e de dar ênfase ao procedimento. Nessas equações, o sinal + refere-se à soma matricial ou vetorial após a projeção dos elementos envolvidos para o mesmo espaço. Isso será feito ao longo de todo o texto, seguindo a literatura sobre o assunto (Whitcomb, 1991).

Nesse contexto, a equação de governo do modelo global é

$$(\mathbf{K}_{GC} + \mathbf{K}_{GL})\mathbf{u}_G = \mathbf{f}_G, \tag{2.30}$$

em que \mathbf{f}_G é o vetor de forças associado ao problema global.

O campo de deslocamentos \mathbf{u}_G obtido por meio do sistema de equações 2.30 é a solução global inicial, denotada por \mathbf{u}_G^0 . Tal solução é transferida como condição de contorno de Dirichlet para o problema local. A aproximação local \mathbf{u}_L é, então, obtida por meio da equação

$$\mathbf{K}_L \mathbf{u}_L = \mathbf{f}_L. \tag{2.31}$$

Os modelos global e local representam $\bar{\Omega}_L$ de maneira distinta, já que, em geral, $\mathbf{K}_{GL} \neq \mathbf{K}_L$. Uma vez que os deslocamentos \mathbf{u}_G^0 (que dependem de \mathbf{K}_{GL}) foram utilizados

como condição de contorno para a solução de \mathbf{u}_L (que dependem de \mathbf{K}_L). Há, portanto, compatibilidade de deslocamentos na interface, mas não há equilíbrio de forças em Γ .

Por meio dessa observação, Whitcomb (1991) propôs uma estratégia que busca o equilíbrio em Γ por meio de correções no vetor de forças do modelo global. Inicialmente, computa-se o vetor de forças residuais, denotado \mathbf{R} . Esse vetor representa a diferença da reação calculada em Γ pelo modelo global e pelo modelo local. Assim sendo, o vetor de forças residuais é expresso por

$$\mathbf{R} = -(\mathbf{f}_C^\Gamma + \mathbf{f}_L^\Gamma) = -[(\mathbf{K}_{GC}\mathbf{u}_{GC} - \mathbf{f}_{GC}) + (\mathbf{K}_L\mathbf{u}_L - \mathbf{f}_L)]|_\Gamma, \quad (2.32)$$

em que o sinal negativo é utilizado pois \mathbf{R} é definido como uma reação. É importante observar que a Eq. (2.32) ocorre em Γ . Isto é, a Eq. (2.32) envolve os elementos do modelo global que pertencem à $\bar{\Omega}_C$ e que são adjacentes à Γ , bem como os elementos do modelo local que são adjacentes à Γ .

O vetor de forças do modelo global é, então, atualizado somando-se \mathbf{R} ao vetor inicial. $\underline{\mathbf{R}}$ é o prolongamento de \mathbf{R} para o espaço do modelo global, ou seja, $\underline{R}_i = R_i$ se i refere-se a um nó localizado em Γ . Caso contrário, $\underline{R}_i = 0$. A atualização do modelo global levando em conta o vetor de forças residuais busca *corrigir* a resposta global, de modo que os deslocamentos calculados passem a considerar os efeitos que foram contemplados apenas no modelo local.

O equilíbrio é atingido por um processo iterativo, o que deu origem ao nome do método. O procedimento proposto por Whitcomb pode ser descrito pelas seguintes etapas:

Passo 1 *Análise global*: resolver o modelo global e obter a solução \mathbf{u}_G^k , conforme Eq. (2.30). A primeira iteração é $k = 0$, de modo que a solução global inicial é denotada \mathbf{u}_G^0 .

Passo 2 *Atualização dos modelos locais*: utilizar a solução do modelo global, \mathbf{u}_G^k , para definir as condições de contorno de Dirichlet dos modelos locais.

Passo 3 *Análises locais*: resolver todos os modelos locais e obter a solução \mathbf{u}_L^k para cada um deles, conforme Eq. (2.31). Quando houver mais de um modelo local, as respectivas soluções podem ocorrer em paralelo.

Passo 4 *Forças residuais*: utilizar as soluções do modelo global e dos modelos locais para calcular o vetor de forças residuais \mathbf{R} por meio da Eq. (2.32).

Passo 5 *Convergência*: verificar se houve convergência da solução. Será necessário, portanto, estabelecer um critério de convergência. Esse assunto será tratado na Seção 3.1. Em caso de atendimento ao critério de convergência, \mathbf{u}_G^k e \mathbf{u}_L^k são as soluções finais e a análise está concluída. Caso contrário, o procedimento prossegue.

Passo 6 *Atualização do modelo global*: atualizar o vetor de forças do modelo global somando-se o resíduo $\underline{\mathbf{R}}$. Retorna-se, então, ao Passo 1, dando início a iteração $k = k + 1$.

Retomando-se a Fig. 2.9, apresentada na Seção 2.3.3, a equação de governo do problema de referência para o MEF-GLI é

$$(\mathbf{K}_{GC} + \mathbf{K}_L)\mathbf{u}_{ref} = \mathbf{f}, \quad (2.33)$$

de modo que, em suma, o MEF-GLI busca a solução da Eq. (2.33) por um processo iterativo que tem como ponto de partida a solução das Eqs. (2.30) e (2.31). Na hipótese de convergência, a solução aproximada obtida pelo procedimento é construída com os deslocamentos \mathbf{u}_{GC} em $\bar{\Omega}_C$ e com os deslocamentos \mathbf{u}_L em $\bar{\Omega}_L$. Isto é, ao longo das iterações do MEF-GLI, ocorre um processo de substituição da solução \mathbf{u}_{GL} pela solução \mathbf{u}_L — ambas referentes à $\bar{\Omega}_L$. Por essa razão, na Eq. (2.33), a solução \mathbf{u}_{ref} é dita solução de referência do MEF-GLI.

Outro ponto que merece destaque na Eq. (2.33) refere-se à matriz $(\mathbf{K}_{GC} + \mathbf{K}_L)$. No contexto do MEF-GLI, a Eq. (2.33) não é resolvida e essa matriz nunca é, de fato, construída. Decerto, as parcelas que a compõem referem-se à modelos computacionais distintos. Em vez disso, o acoplamento entre os modelos se dá pela transferência de condições de contorno: deslocamentos nodais, que são transferidos do modelo global para os modelos locais (Passo 2 do procedimento), e forças nodais, obtidas a partir do desequilíbrio em Γ (Passo 6). Li, O'Hara e Duarte (2021) comentam que essas são informações básicas e que são comumente obtidas nos arquivos de saída dos programas de análise pelo MEF. Por essa razão, o método é não intrusivo e permite que os modelos global e local sejam tratados por estratégias de solução (*solvers*) distintas sem que sejam necessárias modificações nos códigos-fonte.

A solução obtida pelo MEF-GLI, na hipótese da convergência, é tal que há compatibilidade de deslocamentos e equilíbrio de forças em Γ . A compatibilidade de deslocamentos é satisfeita utilizando-se a solução de um modelo como condição de contorno de Dirichlet do outro (Passo 2). Já o equilíbrio de forças é o objetivo do algoritmo, sendo atingido por meio do decréscimo dos valores de \mathbf{R} até um valor adequado, a ser relacionado com o critério de convergência.

Existem duas limitações associadas ao MEF-GLI que merecem destaque:

1. A taxa de convergência do procedimento depende da *diferença de rigidez*³ entre \mathbf{K}_{GL} e \mathbf{K}_L . O procedimento herda essa característica por ser não intrusivo, conforme comentado na Seção 2.3.3.
2. A eficiência computacional e a facilidade de implementação do método estão fortemente atreladas à conformidade das malhas dos modelos em Γ .

Em relação ao primeiro ponto, os problemas ocorrem quando os modelos global e local apresentam significativa *diferença de rigidez* na discretização do domínio local. A seção

³Ver comentário da página 44 a respeito do termo.

Seção 3.2.2 explora o comportamento da convergência do método de forma mais detalhada. Por ora, é importante comentar que a situação mais comum ocorre quando o problema local é significativamente mais flexível do que o global, o que pode ser representado, sem rigor matemático, por $\mathbf{K}_L \ll \mathbf{K}_{GL}$. Nesse cenário, a convergência sempre ocorre, porém, em taxa tão mais lenta quanto maior a diferença entre as respostas iniciais obtidas por \mathbf{K}_{GC} e \mathbf{K}_L . O problema é agravado quando $\mathbf{K}_L > \mathbf{K}_{GL}$, isto é, quando a discretização de $\bar{\Omega}_L$ é mais rígida no modelo local do que no modelo global. A partir de certo ponto, não haverá convergência e não será obtida uma solução que satisfaça o equilíbrio de forças em Γ . Em ambos os casos, é possível alcançar e/ou acelerar a convergência pelo uso de algoritmos numéricos mais refinados (Allix e Gosselet, 2020). Uma discussão acerca desses algoritmos é apresentada na Seção 3.3.

Quanto ao segundo ponto, Whitcomb (1991) comenta que é necessário aplicar estratégias especiais para o acoplamento nas situações em que as malhas dos modelos global e local são não conformes em Γ . Cita-se, como exemplo, o uso de multiplicadores de Lagrange – abordagem discutida na Seção 2.3.1. Em geral, o uso dessas estratégias especiais implica em aumento do custo computacional do acoplamento, bem como introduz significativas dificuldades no processo de implementação do acoplamento. Por outro lado, a construção de malhas conformes nas interfaces entre o modelo global e os modelos locais também pode ser uma tarefa trabalhosa e, com frequência, limitante do processo.

2.5 Acoplamento não intrusivo para o MEF gl

Algoritmos para o acoplamento não intrusivo envolvendo o MEF gl têm sido tema de trabalhos recentes no ambiente acadêmico. Em geral, as abordagens baseiam-se na propriedade hierárquica do enriquecimento global-local, que garante que as funções de forma padrão do MEF, definidas antes do enriquecimento global-local, permaneçam inalteradas após o enriquecimento (Duarte e Kim, 2008). Tal propriedade é evidenciada na Eq. (2.7). É possível dissociar a contribuição da solução oriunda do MEF daquela proveniente do enriquecimento e, assim, realizar o acoplamento entre a solução inicial e a solução pelo MEF gl de forma não intrusiva.

Gupta, Pereira, Kim, Duarte e Eason (2012) apresentaram um método de acoplamento não intrusivo para a solução de problemas de mecânica da fratura no espaço tridimensional. Nessa abordagem, o problema global enriquecido é solucionado por meio de iterações não intrusivas entre duas estratégias de solução distintas (*solvers*). O problema não enriquecido é solucionado por um programa de análise pelo MEF, enquanto um *software* MEF gl é responsável pela análise local e solução do problema global enriquecido. Em abordagem similar, Plews, Duarte e Eason (2012) solucionaram problemas de elevados gradiente térmicos. Em ambas as propostas, o acoplamento é não intrusivo já que somente os vetores de forças e de deslocamentos nodais são responsáveis pela troca de informações

entre os *solvers*. Também não há necessidade de se realizar qualquer modificação nos códigos-fonte dos programas computacionais.

Nessas estratégias, contudo, o acoplamento entre os modelos é realizado pela condensação estática do sistema de equações do problema global, Eq. (2.12). Para tal, é necessário acesso às matrizes de rigidez dos problemas e, no contexto de pacotes computacionais comerciais, isso não é possível na maioria dos casos. Ainda, a determinação do vetor de forças requer o manuseio de vetores com dimensão da ordem de grandeza do problema global. Como consequência, o consumo de memória pode ser elevado e a abordagem pode se mostrar onerosa para aplicação em problemas de escala industrial, com elevado número de graus de liberdade. Essas características, portanto, consistem limitações dessas abordagens tendo em vista o seu uso em escala industrial.

Posteriormente, Fillmore e Duarte (2018) apresentaram uma alternativa de acoplamento não intrusivo para o MEF G^{gl} que não utilizava a condensação estática do sistema de equações do problema global enriquecido. Nessa versão, batizada como *Hierarchical Non-Intrusive Algorithm* (HNA), é necessário que o *software* de análise pelo MEF seja capaz de exportar a matriz de rigidez e o vetor de forças do problema global. Essas informações são importadas pelo programa de análise pelo MEF G . O método reduz a demanda computacional da solução, porém, requer que o tipo de elemento finito utilizado esteja presente na biblioteca de elementos dos dois *softwares* envolvidos. Além disso, persiste a necessidade de acesso à matriz de rigidez do modelo global.

Finalmente, Li, O'Hara e Duarte (2021) introduziram o MEF G^{gl} -GLI. A abordagem recorre ao procedimento apresentado na Seção 2.4 e não apresenta as limitações mencionadas para os outros métodos citados nesta seção. Exige-se apenas a transferência de informações básicas: deslocamentos e forças generalizadas. Essa estratégia, cuja implementação foi o tema principal deste trabalho, é apresentada a seguir.

2.5.1 MEF G^{gl} e algoritmo Global-Local Iterativo

A estratégia para o acoplamento não intrusivo proposta por Li, O'Hara e Duarte (2021) combina o procedimento global-local iterativo com o MEF G^{gl} . Por essa razão, será aqui denominada MEF G^{gl} -GLI, em detrimento à abreviação originalmente utilizada pelos autores, decorrente do nome do método na língua inglesa. A estratégia é voltada para problemas multiescala e permite o uso conjunto de diferentes (*solvers*). Isso possibilita, por exemplo, a associação de programas computacionais voltados à indústria com programas desenvolvidos em ambiente acadêmico. A principal limitação desse método estaria associada ao comportamento da convergência, pois as características do GLI discutidas na Seção 2.4 são aqui herdadas. O MEF G^{gl} -GLI pode ser aplicado em análises multifísicas e sua implementação é relativamente simples, tendo em vista a complexidade do problema abordado. A expectativa é que o método apresente boa eficiência computacional

na solução de problemas de escala industrial com comportamentos localizados de maior complexidade.

No MEF G^{gl} -GLI, o problema é subdividido em três escalas. A *escala global* é a que abrange todo o problema. Ela é discretizada de tal forma que não haverá preocupação em se representar adequadamente efeitos localizados. O respectivo modelo recebe, portanto, uma malha grosseira.

A *mesoescala* é uma região contida na escala global. Sua principal função é permitir o acoplamento não intrusivo entre as escalas global e local, que será definida adiante. A interface entre a mesoescala e a escala global é denotada por Γ . A mesoescala deve ser construída de modo a possuir em Γ a mesma discretização utilizada na escala global. Assim, uma vez que está garantida a conformidade de malhas em Γ , não é necessário utilizar técnicas especiais para o acoplamento de malhas não conformes. Evita-se, dessa forma, uma das principais limitações do método proposto por Whitcomb (1991). Ressalta-se que, a princípio, a mesoescala também não contempla efeitos locais.

Por fim, são definidas uma ou mais regiões locais, denominadas escala local. Nessa escala, ocorre a modelagem detalhada do problema e todos os fenômenos localizados de interesse são considerados. Os modelos da escala local são elaborados com malha adequadamente refinada, isto é, compatível com o problema avaliado.

Quanto à interação entre as escalas, o GLI realiza a troca de informações entre a escala global e a mesoescala, enquanto o MEF G^{gl} realiza o enriquecimento global-local da mesoescala a partir da solução da escala local. O fluxo do procedimento está apresentado na Fig. 2.11. A análise MEF G^{gl} -GLI pode ser descrita pelas seguintes etapas:

Passo 1 *Análise global*: resolver o modelo global e obter a solução \mathbf{u}_G^k do problema. Na primeira iteração, $k = 0$ e \mathbf{u}_G^0 é a solução global inicial.

Passo 2 *Atualização e análise da mesoescala*: definir as condições de contorno de Dirichlet dos modelos da mesoescala a partir da solução \mathbf{u}_G^k . Resolver os modelos da mesoescala e obter a solução \mathbf{u}_M^k para cada um deles.

Passo 3 *Atualização e análise da escala local*: definir as condições de contorno dos modelos da escala local a partir da solução \mathbf{u}_M^k . Um modelo da mesoescala pode estar associado a mais de um modelo local. Resolver os modelos da escala local e obter a solução \mathbf{u}_L^k para cada um deles.

Passo 4 *Análise mesoescala enriquecida*: construir funções para o enriquecimento dos modelos da mesoescala a partir das soluções \mathbf{u}_L^k (enriquecimento global-local). Resolver os modelos enriquecidos da mesoescala e obter a solução \mathbf{u}_{ME}^k .

Passo 5 *Vetor de forças residuais*: calcular o vetor de forças residuais \mathbf{R}^k por meio da Eq. (2.32). Isto é feito utilizando as soluções \mathbf{u}_G^k e \mathbf{u}_{ME}^k .

Passo 6 *Convergência*: verificar se há convergência da solução. Nesse caso, \mathbf{u}_G^k , \mathbf{u}_{ME}^k

e \mathbf{u}_L^k são as soluções finais. Caso a convergência não tenha sido atingida, prosseguir para o próximo passo.

Passo 7 Atualização da escala global: atualizar o vetor de forças do modelo global somando-se o resíduo \mathbf{R}^k . Retornar para o Passo 1, dando início a iteração $k = k + 1$.

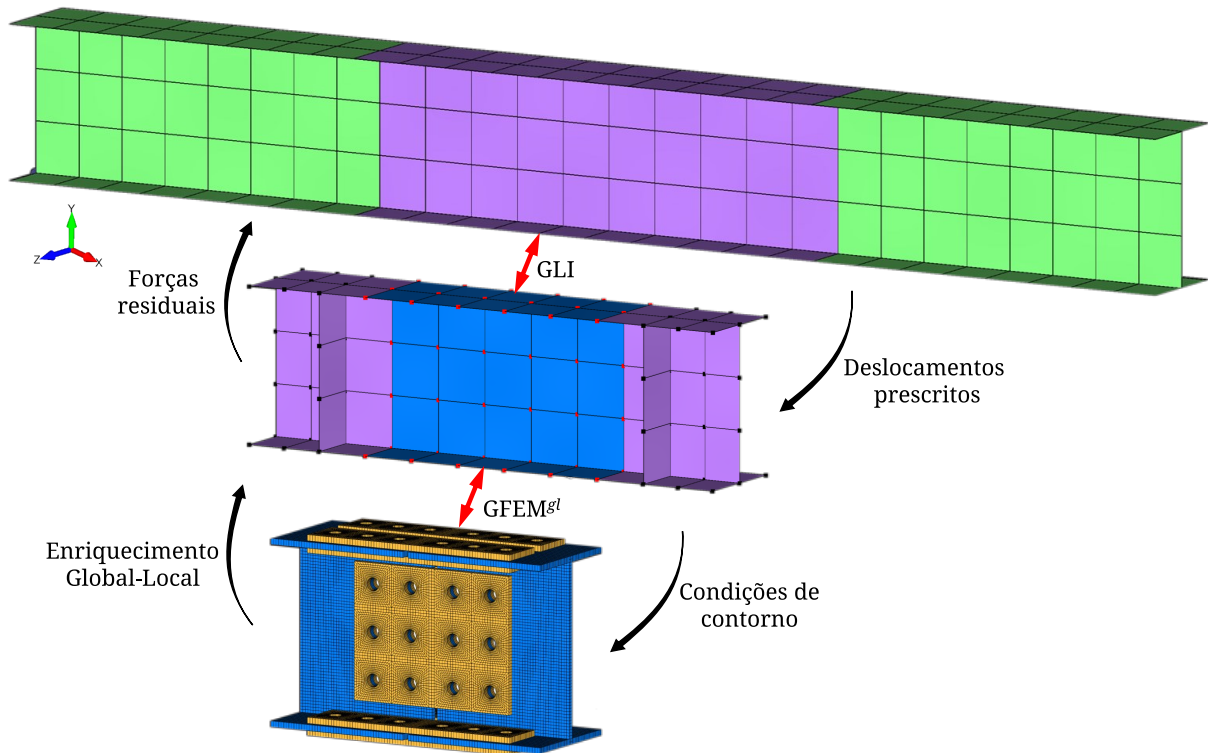


Figura 2.11: Processo de solução pelo MEFG^{gl} -GLI.

No procedimento descrito, tal como ilustrado na Fig. 2.11, observa-se que a mesoescala possui um papel duplo. Ela é a escala local de uma análise global-local que utiliza o algoritmo GLI, mas também é a escala global de um problema resolvido pelo MEFG^{gl} .

O modelo da escala global é resolvido pelo MEF. Dadas as características dessa escala, o uso de um programa com elevada eficiência é interessante. A mesoescala e a escala local constituem um problema resolvido pelo MEFG^{gl} e, assim, são avaliadas por programa desenvolvido em ambiente de pesquisa, uma vez que essa estratégia de solução ainda não está disponível na indústria.

Capítulo 3

Aspectos computacionais do MEFG^{gl}-GLI

Este capítulo se destina à apresentação das implementações realizadas neste trabalho. Inicialmente, é apresentado um algoritmo para o MEFG^{gl}-GLI. Realiza-se uma discussão sobre o comportamento da convergência do algoritmo GLI. Em seguida, são apresentadas algumas técnicas para a aceleração da convergência do método. Por fim, a implementação realizada é apresentada em detalhes.

3.1 Algoritmo

Conforme definido no capítulo anterior, o MEFG^{gl}-GLI, proposto por Li, O’Hara e Duarte (2021), é um método não intrusivo para a solução numérica em três escalas que associa duas estratégias de solução. A mesoescala e a escala local são solucionadas por meio da estratégia global-local aplicada ao MEFG, isto é, pelo MEFG^{gl}. Por sua vez, a escala global global resolvida pelo MEF. O acoplamento entre as soluções da escala global e a mesoescala enriquecida é realizado pela estratégia global-local iterativa, GLI. No procedimento, o acoplamento é realizado trocando-se informações entre as escalas. As informações trocadas são deslocamentos e esforços generalizados.

Os passos que constituem o procedimento, apresentados na Seção 2.5.1 de forma conceitual, são utilizados para a construção do Algoritmo 3.1.

Os modelos computacionais são dados de entrada. Existem no mínimo três modelos, um para cada uma das escalas do problema. Há, portanto, a necessidade de se identificar os arquivos que representam esses modelos de acordo com os programas computacionais envolvidos. Também é necessário identificar, de alguma forma, os nós e elementos da interface entre o modelo global e o modelo da mesoescala (Γ). A Fig. 3.1 esclarece quais são esses nós e elementos.

Após o recebimento dos dados de entrada, o algoritmo consiste na repetição dos pas-

Algoritmo 3.1: Solução pelo MEF G^{gl} -GLI.**Entrada:**

- Modelos numéricos da escala global, da mesoescala e da escala local.
- Lista dos nós da interface entre os modelos global e meso.
- Lista dos elementos da interface entre os modelos global e meso.

Saída:

- Solução final da escala global: \mathbf{u}_G .
- Solução final da mesoescala enriquecida: $\mathbf{u}_{ME,n_{GL}}$.
- Solução final da escala local: $\mathbf{u}_{L,n_{GL}}$.

```

1 enquanto verdadeiro faça
2   | Calcular a solução do modelo global:  $\mathbf{u}_G$ ;
3   | Impor  $\mathbf{u}_G$  como condição de contorno em  $\Gamma$  na mesoescala;
4   | Calcular a solução inicial do modelo da mesoescala:  $\mathbf{u}_M$ ;
5   | para  $j = 0$  até  $n_{GL}$  faça
6     |   | Calcular a solução do modelo local:  $\mathbf{u}_{L,j}$ ;
7     |   | Enriquecer o modelo da mesoescala a partir da solução  $\mathbf{u}_{L,j}$ ;
8     |   | Calcular a solução do modelo enriquecido da mesoescala:  $\mathbf{u}_{ME,j}$ ;
9   | fim
10  | Calcular  $\mathbf{R}$  pela Eq. (2.32) utilizando  $\mathbf{u}_G$  e  $\mathbf{u}_{ME,n_{GL}}$ ;
11  | se há convergência então
12  |   | interromper;
13  | senão
14  |   | Atualizar o vetor de forças do modelo global:  $\mathbf{f}_G = \mathbf{f}_G + \mathbf{R}$ ;
15  | fim
16 fim

```

sos enumerados ao final da Seção 2.5.1 até que o critério de convergência seja atingido. Inicialmente, procede-se a solução inicial do problema global \mathbf{u}_G , conforme a Linha 2 do Algoritmo 3.1. Em geral, o modelo global é extenso e a obtenção da sua solução pode ser onerosa. Tendo em vista que esse modelo trata de efeitos com comportamento globais, é interessante que sua solução seja obtida por meio de programas computacionalmente eficientes, como, por exemplo, as implementações do MEF difundidas na indústria — por exemplo, *softwares* como Abaqus, ANSYS, Nastran e ADINA. Na implementação realizada neste trabalho, foi utilizado o Abaqus.

Em seguida, a solução \mathbf{u}_G é utilizada para a definição de deslocamentos prescritos no modelo da mesoescala. Essa transferência ocorre somente nos nós situados em Γ e, portanto, demanda-se a correspondência entre os nós dos modelos da escala global e da mesoescala. Isto é, para cada nó \mathbf{x}_j^Γ do modelo global que pertence à Γ , deve-se identificar qual é o seu correspondente no modelo da mesoescala. Recordar-se que, na forma como o método foi proposto por Li, O'Hara e Duarte (2021), as malhas do modelo

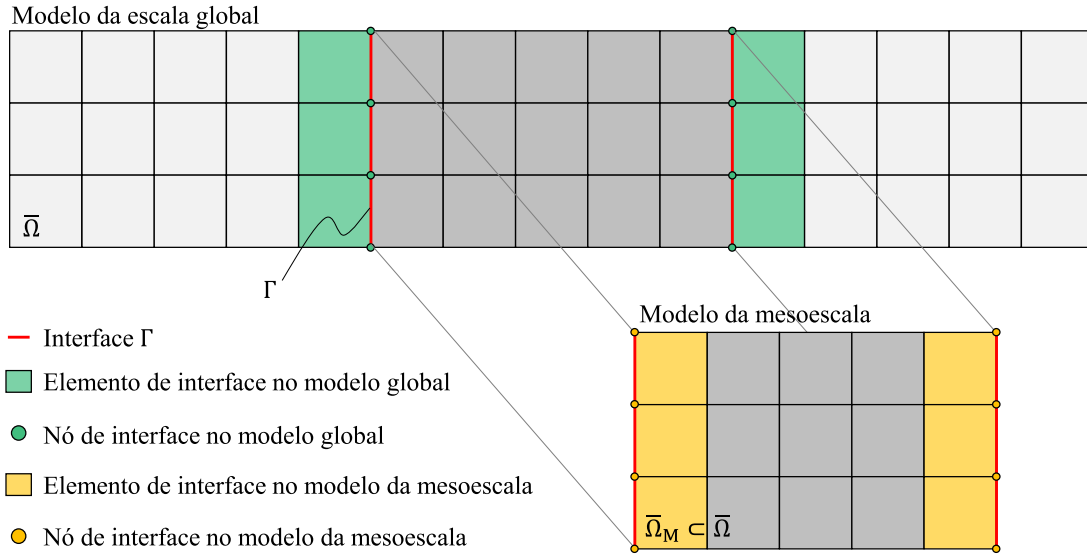


Figura 3.1: Identificação dos nós e dos elementos de interface no MEF G^{gl} -GLI.

da escala global e da mesoescala são conformes, o que significa que cada nó \mathbf{x}_j^Γ possui um, e somente um, nó correspondente na mesoescala.

Uma vez que o modelo da mesoescala está preparado, inicia-se o procedimento de solução pelo MEF G^{gl} . Essa análise corresponde às Linhas 4 a 9 do Algoritmo 3.1 e, assim, deve ser realizada por um *software* que tenha implementado o MEF G^{gl} . Na data de publicação deste trabalho, os programas mais difundidos na indústria não oferecem tal implementação, de modo que a análise pelo MEF G^{gl} é realizada por um *software* desenvolvido em ambiente de pesquisa. Nesta implementação, foi utilizado o **INSANE**. Ainda em relação à análise MEF G^{gl} , comenta-se que podem existir iterações entre a mesoescala e a escala local. No Algoritmo 3.1, esse número de iterações é representado por n_{GL} .

Na Linha 10, computa-se o vetor de forças residuais. Isso é feito utilizando-se as soluções \mathbf{u}_G e \mathbf{u}_{ME} na Eq. (2.32). Calcula-se, assim, as reações nos elementos da interface Γ tanto no modelo global quanto no modelo da mesoescala. Destaca-se que, para o modelo global, tais elementos estão localizados em $\bar{\Omega}_C$. A Fig. 3.1 indica quais são os elementos envolvidos no cálculo do vetor de forças residuais.

Verifica-se, então, o atendimento ao critério de convergência. Whitcomb (1991) sugere que seja definido um valor de tolerância admissível. Identifica-se o maior valor absoluto dos elementos que constituem o vetor \mathbf{R} e a convergência seria atingida quando tal valor fosse inferior à tolerância. Em seu trabalho, Whitcomb indica que tolerâncias inferiores a 10^{-7} não são eficientes, pois adicionam muitas iterações ao processo e não proporcionam melhoria aos resultados. Existem, contudo, alguns problemas associados ao critério descrito. O mais evidente é o critério ser estabelecido a partir de uma tolerância absoluta predeterminada. Isto é, a tolerância não é relativa, o que significa que o número de iterações até a convergência depende da magnitude dos esforços atuantes. Além disso,

a comparação do valor de tolerância com o valor de apenas um dos elementos do vetor \mathbf{R} pode não ser representativa. Por essa razão, optou-se pelo critério de convergência a seguir, apresentado por Li, O'Hara e Duarte (2021):

$$\frac{\|\mathbf{R}^k\|}{\|\mathbf{R}^0\|} < \epsilon, \quad (3.1)$$

em que $\|\mathbf{R}^0\|$ é a norma euclidiana do vetor \mathbf{R} na primeira iteração ($k = 0$), enquanto $\|\mathbf{R}^k\|$ refere-se à iteração k . Neste texto, a razão apresentada no lado da esquerda da Eq. (3.1) será muitas vezes referida como *resíduo relativo*. A convergência é atingida quando o resíduo relativo é inferior à ϵ , um valor predefinido de tolerância.

O processo descrito nesta seção indica que a implementação do MEF gl -GLI realizada neste trabalho possui três atores distintos. A Tabela 3.1 busca evidenciar qual código é responsável pela realização de cada uma das etapas do Algoritmo 3.1. Há menção às técnicas de aceleração da convergência, que serão discutidas na Seção 3.3 e, portanto, não constam no Algoritmo 3.1.

Tabela 3.1: Códigos responsáveis pelas etapas do MEF gl -GLI.

Linha	Etapa	Código responsável
Entrada	Leitura dos dados iniciais	Acoplamento
2	Solução da escala global	MEF
3	Atualização do modelo da mesoescala	Acoplamento
4 a 9	Solução da mesoescala e da escala local	MEF gl
10	Cálculo do vetor de forças residuais	
11	Teste de convergência	
14	Atualização do modelo da escala global	Acoplamento
Saída	Obtenção das soluções finais	
–	Técnicas de aceleração	

3.2 Convergência do GLI

Na seção Seção 2.5.1 foram comentadas duas limitações notáveis do GLI. A proposição da mesoescala garante a conformidade entre as malhas dos dois problemas acoplados pelo GLI, tratando, assim, a primeira limitação. A outra limitação diz respeito às propriedades de convergência do método e será explorada nessa seção.

Inicialmente, o Algoritmo 3.1 é reescrito em sua forma incremental. Esse formato é

mais conveniente por algumas razões. O estudo do comportamento da convergência do GLI torna-se mais claro. Mais importante, a forma incremental do problema permite o uso de técnicas de aceleração da convergência (Duval, Passieux, Salaün e Guinard, 2014). Em seguida, estuda-se o comportamento da convergência do problema. E, então, são apresentadas técnicas para a aceleração da convergência. Algumas dessas técnicas foram implementadas e aplicadas na solução de alguns problemas, o que está apresentado na Seção 4.3.4.

Destaca-se que as equações a seguir são apresentadas com a mesma notação utilizada no Capítulo 2. O símbolo $_$ abaixo de vetores e matrizes indica operação de prolongamento, isto é, esses elementos devem ser multiplicados por matrizes booleanas construídas para prolongar as informações de uma matriz ou vetor para outra matriz ou vetor de dimensões diferentes.

Faz-se necessário endossar que esta seção se refere ao GLI, isto é, ao procedimento proposto por Whitcomb (1991) para o acoplamento não intrusivo entre duas escalas de solução. Nesse sentido, as duas escalas são denominadas escala global e escala local. Essa terminologia não deve ser confundida com a adotada no MEF G^{gl} -GLI, procedimento discutido na Seção 2.5.1 e que incorpora o GLI. No MEF G^{gl} -GLI, a mesoescala exerce a função de escala local do GLI. Também é importante lembrar que no MEF G^{gl} -GLI as escalas são construídas de modo a garantir a conformidade de malhas na interface Γ e isso será levando em consideração a seguir.

3.2.1 Forma incremental do problema global

O desenvolvimento matemático apresentado nessa seção está baseado no trabalho de Duval, Passieux, Salaün e Guinard (2014). Seja o problema da Fig. 3.2, cuja imposição do equilíbrio em $\bar{\Omega}$, em uma iteração k , conduz à

$$\mathbf{K}_G \mathbf{u}_G^k = \mathbf{f}_G. \quad (3.2)$$

Pode-se definir as reações na interface Γ do modelo global considerando os domínios $\bar{\Omega}_C$ e $\bar{\Omega}_L$, contidos em $\bar{\Omega}$, pelas respectivas equações

$$\boldsymbol{\eta}_{GC}^k = \left(\mathbf{K}_{GC} \mathbf{u}_{GC}^k - \mathbf{f}_{GC} \right) |_{\Gamma} \quad (3.3)$$

e

$$\boldsymbol{\eta}_{GL}^k = \left(\mathbf{K}_{GL} \mathbf{u}_{GL}^k - \mathbf{f}_{GL} \right) |_{\Gamma}. \quad (3.4)$$

Uma vez que $\bar{\Omega}_C$ e $\bar{\Omega}_L$ fazem parte do mesmo problema (o problema global), é necessário que Γ esteja em equilíbrio. Tem-se, então, que

$$\boldsymbol{\eta}_{GC}^k + \boldsymbol{\eta}_{GL}^k = \mathbf{0}. \quad (3.5)$$

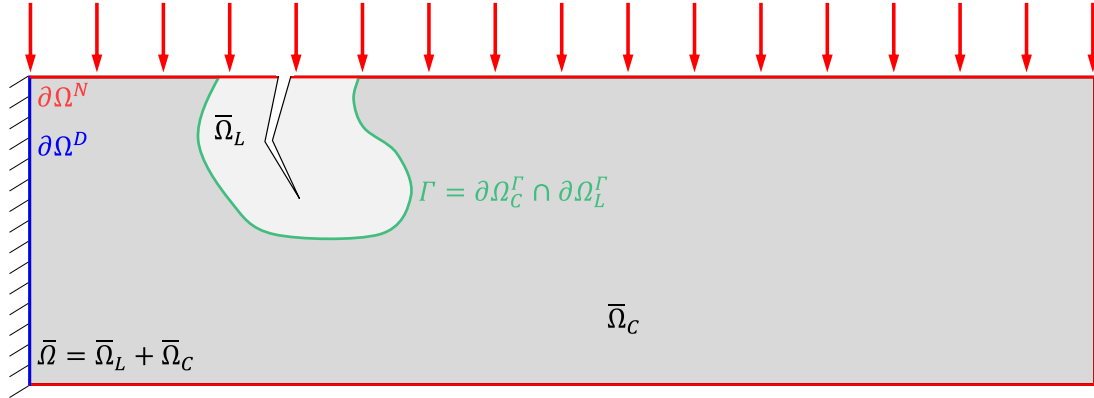


Figura 3.2: Problema geral de um acoplamento não intrusivo.

A Eq. (3.2) pode, assim, ser reescrita sem prejuízo como

$$\mathbf{K}_G \mathbf{u}_G^k = \mathbf{f}_G + \underline{\boldsymbol{\eta}}_{GC}^k + \underline{\boldsymbol{\eta}}_{GL}^k. \quad (3.6)$$

Retoma-se agora o procedimento do GLI, no qual o modelo global receberá o vetor de forças residuais como correção devido aos efeitos localizados. Conforme a Eq. (2.32) e utilizando a definição da Eq. (3.3), o vetor de forças residuais na iteração k do procedimento de análise pelo GLI é

$$\mathbf{R}^k = -(\boldsymbol{\eta}_{GC}^k + \boldsymbol{\eta}_L^k), \quad (3.7)$$

em que $\boldsymbol{\eta}_L^k = (\mathbf{K}_L \mathbf{u}_L^k - \mathbf{f}_L) |_{\Gamma}$ é a reação do modelo local em Γ . Uma vez que o modelo global deve ser atualizado com \mathbf{R}^k , a solução global da próxima iteração, \mathbf{u}_G^{k+1} , é obtida acrescentando-se o vetor de forças residuais na Eq. (3.6), o que conduz à

$$\mathbf{K}_G \mathbf{u}_G^{k+1} = \mathbf{f}_G + \underline{\boldsymbol{\eta}}_{GC}^k + \underline{\boldsymbol{\eta}}_{GL}^k + \mathbf{R}^k. \quad (3.8)$$

A Eq. (3.8) pode ser transformada em um problema incremental por meio de manipulação algébrica. Subtraindo-se o termo $\mathbf{K}_G \mathbf{u}_G^k$ em ambos os lados dessa equação, tem-se

$$\begin{aligned} \mathbf{K}_G \mathbf{u}_G^{k+1} - \mathbf{K}_G \mathbf{u}_G^k &= \mathbf{f}_G + \underline{\boldsymbol{\eta}}_{GC}^k + \underline{\boldsymbol{\eta}}_{GL}^k + \mathbf{R}^k - \mathbf{K}_G \mathbf{u}_G^k \\ \mathbf{K}_G (\mathbf{u}_G^{k+1} - \mathbf{u}_G^k) &= \mathbf{f}_G + \underline{\boldsymbol{\eta}}_{GC}^k + \underline{\boldsymbol{\eta}}_{GL}^k - \underline{\boldsymbol{\eta}}_{GC}^k - \underline{\boldsymbol{\eta}}_L^k - \mathbf{K}_G \mathbf{u}_G^k \\ \mathbf{K}_G (\mathbf{u}_G^{k+1} - \mathbf{u}_G^k) &= \mathbf{f}_G + \underline{\boldsymbol{\eta}}_{GL}^k - \underline{\boldsymbol{\eta}}_L^k - \mathbf{K}_G \mathbf{u}_G^k. \end{aligned} \quad (3.9)$$

O algoritmo é não intrusivo, portanto, \mathbf{K}_G não se altera ao longo das iterações. Isto é, \mathbf{K}_G da Eq. (3.9) é o mesmo da Eq. (3.6). Assim, tem-se que

$$\mathbf{f}_G = \mathbf{K}_G \mathbf{u}_G^k - \underline{\boldsymbol{\eta}}_{GC}^k - \underline{\boldsymbol{\eta}}_{GL}^k \quad (3.10)$$

pode ser substituído na Eq. (3.9), o que resulta em

$$\begin{aligned}
\mathbf{K}_G(\mathbf{u}_G^{k+1} - \mathbf{u}_G^k) &= \mathbf{K}_G \mathbf{u}_G^k - \underline{\boldsymbol{\eta}}_{GC}^k - \underline{\boldsymbol{\eta}}_{GL}^k + \underline{\boldsymbol{\eta}}_{GL}^k - \underline{\boldsymbol{\eta}}_L^k - \mathbf{K}_G \mathbf{u}_G^k \\
&= -\underline{\boldsymbol{\eta}}_{GC}^k - \underline{\boldsymbol{\eta}}_L^k \\
&= -(\underline{\boldsymbol{\eta}}_{GC}^k + \underline{\boldsymbol{\eta}}_L^k).
\end{aligned} \tag{3.11}$$

Com o uso da Eq. (3.7), a equação anterior pode ser reescrita em termos do vetor de forças residuais, ou seja,

$$\mathbf{K}_G(\mathbf{u}_G^{k+1} - \mathbf{u}_G^k) = \underline{\mathbf{R}}^k, \tag{3.12}$$

em que, é bom lembrar, $\underline{\mathbf{R}}^k$ consiste na projeção de \mathbf{R}^k para o espaço do problema global.

A Eq. (3.12) estabelece uma relação entre o *acréscimo* da solução calculada para o passo $k + 1$ e a força residual do passo anterior. Mais do que isso, a Eq. (3.12) permite que a solução problema global seja reescrita em sua forma incremental (Duval et al., 2014, Liu et al., 2014), dada por

$$\begin{aligned}
\mathbf{u}_G^{k+1} &= \mathbf{u}_G^k + \mathbf{K}_G^{-1} \underline{\mathbf{R}}^k \\
&= \mathbf{u}_G^k - \mathbf{K}_G^{-1} f(\mathbf{u}_G^k),
\end{aligned} \tag{3.13}$$

em que $f(\mathbf{u}_G^k) = -\underline{\mathbf{R}}^k$.

A Eq. (3.13) evidencia que a estratégia de análise pelo GLI consiste em um método de Newton modificado, pois \mathbf{K}_G é constante ao longo de toda a solução.

3.2.2 Comportamento da convergência

Para realizar um estudo acerca da convergência do GLI, o vetor de forças residuais é reescrito adicionando-se e subtraindo-se o termo $\boldsymbol{\eta}_{GL}^k$ na equação Eq. (3.7),

$$\mathbf{R}^k = -(\boldsymbol{\eta}_{GC}^k + \boldsymbol{\eta}_L^k) + \boldsymbol{\eta}_{GL}^k - \boldsymbol{\eta}_{GL}^k. \tag{3.14}$$

Pela Eq. (3.10), pode-se escrever

$$\underline{\boldsymbol{\eta}}_{GC}^k + \underline{\boldsymbol{\eta}}_{GL}^k = \mathbf{K}_G \mathbf{u}_G^k - \mathbf{f}_G \tag{3.15}$$

e, substituindo na Eq. (3.14), tem-se

$$f(\mathbf{u}_G^k) = -\underline{\mathbf{R}}^k = \mathbf{K}_G \mathbf{u}_G^k - \mathbf{f}_G + \underline{\boldsymbol{\eta}}_L^k - \underline{\boldsymbol{\eta}}_{GL}^k. \tag{3.16}$$

Sejam, então, definidos os complementos de Schur (Gosselet e Rey, 2006) para as matrizes de rigidez correspondentes à $\bar{\Omega}_L$. \mathbf{S}_L é definido para o modelo local e \mathbf{S}_{GL} para o modelo global. Tem-se, assim,

$$\boldsymbol{\eta}_L = \mathbf{S}_L \mathbf{u}_G |_{\Gamma} \tag{3.17}$$

e

$$\boldsymbol{\eta}_{GL} = \mathbf{S}_{GL} \mathbf{u}_G |_{\Gamma}, \quad (3.18)$$

de modo que a Eq. (3.16) pode ser reescrita após essas definições, o que resulta em¹

$$\begin{aligned} f(\mathbf{u}_G^k) &= -\underline{\mathbf{R}}^k = \mathbf{K}_G \mathbf{u}_G^k - \mathbf{f}_G + \mathbf{S}_L \mathbf{u}_G |_{\Gamma} - \mathbf{S}_{GL} \mathbf{u}_G |_{\Gamma} \\ &= \mathbf{K}_G \mathbf{u}_G^k - \mathbf{f}_G + (\mathbf{S}_L - \mathbf{S}_{GL}) \mathbf{u}_G |_{\Gamma}. \end{aligned} \quad (3.19)$$

Já foi comentado que o GLI busca uma solução que garanta tanto a compatibilidade de deslocamentos quanto o equilíbrio de forças em Γ . A convergência da Eq. (3.13), portanto, é satisfeita conforme $\mathbf{R}^k \rightarrow \mathbf{0} \implies f(\mathbf{u}_G^k) \rightarrow \mathbf{0}$. A expressão para a diferenciação de f em relação a \mathbf{u}_G é

$$\nabla f = \mathbf{K}_G + (\mathbf{S}_L - \mathbf{S}_{GL}) \quad (3.20)$$

e permite o estudo do comportamento da convergência do GLI segundo a discretização de Ω_L no modelo global e no modelo local, representadas, na equação, pelos complementos de Schur das suas matrizes de rigidez (Duval, Passieux, Salaün e Guinard, 2014). Quando $\bar{\Omega}_L$ possui rigidez semelhante nos dois modelos, tem-se que $\mathbf{K}_L \approx \mathbf{K}_{GL}$ e, conseqüentemente, \mathbf{S}_L se aproxima de \mathbf{S}_{GL} . Esse cenário implica $\nabla f \approx \mathbf{K}_G$ e pode se dizer que a matriz de rigidez global é uma boa aproximação para ∇f . Esse cenário ocorre, por exemplo, quando o problema local apresenta efeitos de não linearidade material restritos a pequenas regiões do problema.

Quando a discretização de Ω_L é significativamente mais flexível no modelo local, tem-se $\mathbf{K}_L < \mathbf{K}_{GL} \implies \mathbf{S}_L < \mathbf{S}_{GL}$. Nesse caso, verifica-se que \mathbf{K}_G não é uma boa aproximação para ∇f . A convergência ainda ocorre, contudo, pode ser necessário um elevado número de iterações. Esse cenário pode ocorrer quando a escala local considera efeitos que aumentam a magnitude dos deslocamentos obtidos na sua solução em detrimento à solução do modelo global em $\bar{\Omega}_L$. Detalhes geométricos, diferença na descrição do material e alteração na matriz de rigidez do modelo local são alguns exemplos dos efeitos que podem conduzir a esse cenário.

Por fim, pode ocorrer que a representação de $\bar{\Omega}_L$ no modelo local seja mais rígida do que no modelo global. Esse cenário novamente indica que \mathbf{K}_G não é uma boa aproximação para ∇f . Agora, porém, a magnitude de f aumenta a cada iteração e não haverá convergência.

Conforme Liu, Sun e Fan (2014), o comportamento de convergência do GLI indica que as técnicas de aceleração da convergência são necessárias para que sua implementação seja mais eficiente e, principalmente, mais abrangente. Essas técnicas são o assunto da próxima seção.

¹Na Eq. (3.19), os termos que envolvem o produto dos complementos de Schur por \mathbf{u}_G devem ser projetados para o espaço do problema global, de modo a apresentarem a mesma dimensão dos demais termos da equação. A notação utilizada neste texto para tal operação ($|_{\Gamma}$) foi omitida para promover a clareza da equação.

3.3 Técnicas de aceleração

Existem diversas técnicas de aceleração da convergência de problemas como o apresentado na Eq. (3.13). Liu, Sun e Fan (2014) as classificas em dois grupos. Essas técnicas não somente aceleram a convergência da solução, mas também permitem a convergência em casos nos quais a solução convencional iria divergir.

O primeiro grupo é a família de técnicas associada ao método quasi-Newton (Gendre et al., 2009). Em comum, as técnicas pertencentes a essa família envolvem a modificação da matriz de rigidez do problema global. Apesar de melhorar o comportamento da convergência, esse grupo de técnicas pode ter implementação complexa e pode ter alto consumo de memória. Outro ponto importante está na necessidade de acesso à \mathbf{K}_G , o que pressupostamente vai contra a proposta não intrusiva do GLI.

O segundo grupo de técnicas age somente na função $f(\mathbf{u}_G^k)$, mantendo os demais termos da formulação do problema inalterados. Em geral, as técnicas desse grupo buscam realizar uma correção na solução \mathbf{u}_G^k a partir das soluções obtidas nos passos anteriores, isto é, todos os passos que antecederam k . Possuem, em geral, implementação mais simples, ainda que potencialmente eficientes quanto à melhora das propriedades de convergência.

É importante comentar que, segundo Duval, Passieux, Salaün e Guinard (2014), as técnicas apresentadas requerem que o problema de convergência seja abordado na forma incremental, apresentada na Eq. (3.13). A seguir, apresenta-se uma discussão sobre a implementação de algumas dessas técnicas.

3.3.1 Relaxação estática

Essa é a técnica mais simples dentre as disponíveis e sua aplicação no GLI foi prevista já na proposição do método por Whitcomb (1991). Métodos de relaxação têm sido extensivamente utilizados há décadas (Southwell, 1940, 1946). Existem diferentes abordagens e aqui será apresentada a discutida em Xu (1992).

Em uma solução numérica iterativa como a associada à Eq. (3.13), a relaxação é realizada a partir da equação

$$\mathbf{u}_G^{k+1} = \mathbf{u}_G^k + \omega \Delta \mathbf{u}_G^{k+1}, \quad (3.21)$$

em que $\Delta \mathbf{u}_G^{k+1} = g(\mathbf{u}_G^k) - \mathbf{u}_G^k$ e ω é um escalar predefinido chamado fator de relaxação e o seu valor é constante ao longo de todas as iterações. A função g é aquela que calcularia \mathbf{u}_G^{k+1} caso a correção não fosse aplicada. Isto é, g é a própria Eq. (3.13). Essa solução, aqui denominada *solução prevista*, é denotada por $\bar{\mathbf{u}}_G$. Assim, a Eq. (3.21) pode ser escrita como

$$\mathbf{u}_G^{k+1} = \mathbf{u}_G^k + \omega(\bar{\mathbf{u}}_G^{k+1} - \mathbf{u}_G^k). \quad (3.22)$$

Resta, portanto, definir o valor de ω . No contexto do GLI, quando $\bar{\Omega}_L$ apresenta-se mais rígido na discretização da escala local do que na global, situação na qual a convergência não ocorreria, uma solução correta sempre será obtida ao se utilizar $0 < \omega < 1,0$ (Chevreuil, Nouy e Safatly, 2013). Problemas nos quais o modelo local apresenta-se menos rígido do que o global podem, também, ser acelerados. Nestes casos, porém, ω deverá ser maior do que a unidade.

No contexto do MEFG^{gl}-GLI, a aplicação da relaxação exige alterações no Algoritmo 3.1. A principal refere-se à Eq. (3.22) e consiste na correção do vetor solução previsto pelo modelo da escala global, $\bar{\mathbf{u}}_G^k$. O Algoritmo 3.2 apresenta o procedimento do MEFG^{gl}-GLI com relaxação estática.

Observa-se que a relaxação estática é de fácil implementação. Também não há ônus computacional significativo, afinal, a aplicação da técnica requer apenas duas operações entre vetores com dimensão igual ao número de graus de liberdade do modelo global, conforme a Linha 6 do algoritmo.

A limitação da técnica reside na definição de um valor ótimo para o fator de relaxação, ω_{otm} . Chevreuil, Nouy e Safatly (2013) e Xu (1992) apresentam metodologias para a obtenção ou estimação de ω_{otm} , contudo, essa pode ser uma tarefa computacionalmente onerosa, uma vez que as metodologias disponíveis envolvem a manipulação de matrizes de grandes dimensões e extração de autovalores, por exemplo.

3.3.2 Relaxação Dinâmica

A relaxação dinâmica aproveita os conceitos introduzidos na seção anterior e incorpora a ideia de se utilizar um fator de relaxação a cada iteração, isto é,

$$\mathbf{u}_G^{k+1} = \mathbf{u}_G^k + \omega^k (\bar{\mathbf{u}}_G^{k+1} - \mathbf{u}_G^k). \quad (3.23)$$

Há, portanto, a necessidade de se definir uma maneira de atualizar o valor de ω a cada iteração. Existem diferentes formas de se realizar essa tarefa. Irons e Tuck (1969) apresentaram uma versão modificada da fórmula Δ^2 de Aitken (Aitken, 1927), uma técnica de aceleração da convergência de métodos numéricos utilizados na obtenção de raízes de funções. Essa versão foi utilizada por diversos autores. Küttler e Wall (2008) a utilizaram para a solução de problemas de iteração fluido-estrutura. Liu, Sun e Fan (2014) a aplicaram para a construção de uma estratégia global-local para problemas com interfaces não conformes. Duval, Passieux, Salaün e Guinard (2014) a implementaram na solução de problemas multiescala. Li, O'Hara e Duarte (2021) também utilizaram essa metodologia ao propor o MEFG^{gl}-GLI. Em comum, os autores indicam que o método apresenta bom desempenho. Os resultados de Duval, Passieux, Salaün e Guinard (2014) sugerem que o uso da relaxação dinâmica é mais eficiente do que relaxação convencional mesmo quando ω_{otm} é utilizado — ressalta-se que, na relaxação estática, ω_{otm} é calculado

Algoritmo 3.2: Solução pelo MEF G^{gl} -GLI com relaxação estática.

Entrada:

- Modelos numéricos da escala global, da mesoescala e da escala local.
- Lista dos nós da interface entre os modelos global e meso.
- Lista dos elementos da interface entre os modelos global e meso.
- Fator de relaxação ω .

Saída:

- Solução final da escala global: \mathbf{u}_G .
- Solução final da mesoescala enriquecida: $\mathbf{u}_{ME,n_{GL}}$.
- Solução final da escala local: $\mathbf{u}_{L,n_{GL}}$.

```

1   $k = 0$ ;
2  Calcular a solução do modelo global:  $\mathbf{u}_G$ ;
3  enquanto verdadeiro faça
4  | se  $k \neq 0$  então
5  |    $\bar{\mathbf{u}}_G^k = \mathbf{u}_G^{k-1} + \mathbf{K}_G^{-1} \mathbf{R}^k$ ;           /* solução prevista */
6  |    $\mathbf{u}_G^k = \mathbf{u}_G^{k-1} + \omega(\bar{\mathbf{u}}_G^k - \mathbf{u}_G^{k-1})$ ;   /* solução corrigida */
7  | senão
8  |    $\mathbf{u}_G^k = \bar{\mathbf{u}}_G^k$ ;           /* não há correção na primeira iteração */
9  | fim
10  Impor  $\mathbf{u}_G^k$  como condição de contorno em  $\Gamma$  na mesoescala;
11  Calcular a solução inicial do modelo da mesoescala:  $\mathbf{u}_M$ ;
12  para  $j = 0$  até  $n_{GL}$  faça
13  |   Calcular a solução do modelo local:  $\mathbf{u}_{L,j}$ ;
14  |   Enriquecer o modelo da mesoescala a partir da solução  $\mathbf{u}_{L,j}$ ;
15  |   Calcular a solução do modelo enriquecido da mesoescala:  $\mathbf{u}_{ME,j}$ ;
16  | fim
17  Calcular  $\mathbf{R}^k$  pela Eq. (2.32) utilizando  $\mathbf{u}_G^k$  e  $\mathbf{u}_{ME,n_{GL}}^k$ ;
18  se há convergência então
19  |   interromper;
20  | fim
21   $k = k + 1$ ;
22 fim
```

apenas no início do processo de solução e não é atualizado ao longo das iterações.

Nesse sentido, este trabalho implementa a relaxação dinâmica utilizando a fórmula Δ^2 de Aitken conforme proposto por Irons e Tuck (1969). Se os vetores de incremento de deslocamentos a cada iteração k são definidos como

$$\Delta^k = (\bar{\mathbf{u}}_G^k - \mathbf{u}_G^{k-1})|_{\Gamma}, \quad (3.24)$$

então o valor do fator de relaxação na mesma iteração é dado por

$$\omega^k = -\omega^{k-1} \frac{\Delta^{k-1} \cdot (\Delta^k - \Delta^k)}{\|\Delta^k - \Delta^k\|^2}. \quad (3.25)$$

É importante observar que o cálculo de Δ^k envolve somente os graus de liberdade situados na interface Γ . No contexto MEF gl -GLI, Γ está localizada entre a mesoescala e a escala global, cuja malha é grosseira. Uma vez que há conformidade de malha em Γ , a demanda computacional para o cálculo de ω^k é de pouca significância frente ao custo da solução.

Nota-se, também, que o cálculo de ω^k requer o armazenamento de algumas informações. Além do valor corrente do próprio fator de relaxação, é necessário guardar a *solução prevista* dos passos atual e anterior ($\bar{\mathbf{u}}_G^k$ e $\bar{\mathbf{u}}_G^{k-1}$), bem como a *solução corrigida* dos dois últimos passos (\mathbf{u}_G^{k-1} e \mathbf{u}_G^{k-2}).

Como o cálculo de ω^k requer os resultados das duas iterações anteriores, é necessário arbitrar os valores iniciais do procedimento. Nesse sentido, faz-se $\mathbf{u}_G^0 = \bar{\mathbf{u}}_G^0$ e $\mathbf{u}_G^1 = \bar{\mathbf{u}}_G^1$. Isto é, não se realiza a correção da solução nas duas primeiras iterações e $\omega^0 = \omega^1 = 1, 0$. Observa-se, ainda, que ω^k somente é utilizado na correção da solução da próxima iteração, \mathbf{u}_G^{k+1} .

O Algoritmo 3.3 apresenta a solução pelo MEF gl -GLI utilizando a relaxação dinâmica. Observa-se que a implementação da relaxação dinâmica usando a fórmula Δ^2 de Aitken não demanda grandes alterações em relação ao algoritmo que utiliza a relaxação estática.

3.3.3 Quasi-Newton

Conforme discutido no início da Seção 3.3, métodos de aceleração do tipo quasi-Newton são intrusivos, pois modificam a matriz \mathbf{K}_G na Eq. (3.13). É pressuposto que o uso desse tipo de método no âmbito do MEF gl -GLI violaria a proposta não intrusiva do método.

Duval, Passieux, Salaün e Guinard (2014), contudo, apresentaram uma versão não-intrusiva dessa técnica de aceleração. Os autores utilizam a fórmula de atualização simétrica de posto (SR1) da matriz \mathbf{K}_G e a fórmula de Sherman-Morrison para calcular o vetor $\mathbf{u}_G^{k+1} = [\mathbf{K}_G^k]^{-1} \cdot f(\mathbf{u}_G^k)$ de modo iterativo a partir de \mathbf{u}_G^0 , a solução inicial do problema global – observar que aqui a matriz \mathbf{K}_G é diferente a cada iteração. Assim, o incremento de solução a cada iteração é calculado por

$$\left(\mathbf{K}_G^{i+1}\right)^{-1} \underline{\mathbf{R}}^k = \left(\mathbf{K}_G^i\right)^{-1} \underline{\mathbf{R}}^k - \left(\mathbf{K}_G^i\right)^{-1} \underline{\mathbf{R}}^{i+1} \frac{\left(\underline{\mathbf{R}}^{i+1}\right)^\top \left(\mathbf{K}_G^i\right)^{-1} \underline{\mathbf{R}}^i}{\left(\underline{\mathbf{R}}^{i+1}\right)^\top \left[\mathbf{u}_G^i + \left(\mathbf{K}_G^i\right)^{-1} \underline{\mathbf{R}}^{i+1}\right]}, \quad (3.26)$$

em que $\{i \in \mathbb{Z} \mid 0 \leq i < k\}$. A Eq. (3.26) pode, portanto, ser implementada recursivamente e requer o armazenamento de informações de todas as iterações de análise. A

Algoritmo 3.3: Solução pelo MEF G^l -GLI com relaxação dinâmica.

Entrada:

- Modelos numéricos da escala global, da mesoescala e da escala local.
- Lista dos nós da interface entre os modelos global e meso.
- Lista dos elementos da interface entre os modelos global e meso.

Saída:

- Solução final da escala global: \mathbf{u}_G .
- Solução final da mesoescala enriquecida: $\mathbf{u}_{ME,n_{GL}}$.
- Solução final da escala local: $\mathbf{u}_{L,n_{GL}}$.

```

1  $k = 0$ ;
2  $\omega^0 = 1, 0$ ;
3 Calcular a solução do modelo global:  $\mathbf{u}_G$ ;
4 enquanto verdadeiro faça
5     se  $k > 1$  então
6          $\bar{\mathbf{u}}_G^k = \mathbf{u}_G^{k-1} + \mathbf{K}_G^{-1} \mathbf{R}^k$ ;           /* solução prevista */
7         Calcular  $\omega^{k-1}$  pela Eq. (3.25);           /* atualização de  $\omega$  */
8          $\mathbf{u}_G^k = \mathbf{u}_G^{k-1} + \omega^{k-1}(\bar{\mathbf{u}}_G^k - \mathbf{u}_G^{k-1})$ ; /* solução corrigida */
9     senão
10         $\mathbf{u}_G^k = \bar{\mathbf{u}}_G^k$ ;           /* não há correção nas duas primeiras iterações */
11    fim
12    Impor  $\mathbf{u}_G^k$  como condição de contorno em  $\Gamma$  na mesoescala;
13    Calcular a solução inicial do modelo da mesoescala:  $\mathbf{u}_M$ ;
14    para  $j = 0$  até  $n_{GL}$  faça
15        Calcular a solução do modelo local:  $\mathbf{u}_{L,j}$ ;
16        Enriquecer o modelo da mesoescala a partir da solução  $\mathbf{u}_{L,j}$ ;
17        Calcular a solução do modelo enriquecido da mesoescala:  $\mathbf{u}_{ME,j}$ ;
18    fim
19    Calcular  $\mathbf{R}^k$  pela Eq. (2.32) utilizando  $\mathbf{u}_G^k$  e  $\mathbf{u}_{ME,n_{GL}}^k$ ;
20    se há convergência então
21        interromper;
22    fim
23     $k = k + 1$ ;
24 fim

```

saber, as soluções $\bar{\mathbf{u}}_G^i$ e os vetores de forças residuais \mathbf{R}^i devem ser armazenados k vezes, bem como deve ser guardado o produto $(\mathbf{K}_G^i)^{-1} \cdot \mathbf{R}^{k+1}$ para todas as iterações $i < k - 1$. O procedimento para a implementação da técnica de aceleração está apresentado nos Algoritmos 3.4 e 3.5 (Gendre, Allix, Gosselet e Comte, 2009).

Algoritmo 3.4: Solução pelo MEF G^{gl} -GLI com aceleração quasi-Newton.

Entrada:

- Modelos numéricos da escala global, da mesoescala e da escala local.
- Lista dos nós da interface entre os modelos global e meso.
- Lista dos elementos da interface entre os modelos global e meso.

Saída:

- Solução final da escala global: \mathbf{u}_G .
- Solução final da mesoescala enriquecida: $\mathbf{u}_{ME,n_{GL}}$.
- Solução final da escala local: $\mathbf{u}_{L,n_{GL}}$.

```

1  $k = 0$ ;
2 Calcular a solução do modelo global:  $\mathbf{u}_G$ ;
3 enquanto verdadeiro faça
4   se  $k > 1$  então
5      $\Delta \bar{\mathbf{u}}_G^k = \mathbf{K}_G^{-1} \mathbf{R}^k$ ; /* incremento previsto */
6      $\Delta \mathbf{u}_G^k \leftarrow$  Algoritmo 3.5 com  $k$ ,  $\mathbf{R}^k$  e  $\Delta \bar{\mathbf{u}}_G^k$ ; /* incremento corrigido */
7      $\mathbf{u}_G^k = \mathbf{u}_G^{k-1} + \Delta \mathbf{u}_G^k$ ; /* solução corrigida */
8   senão
9      $\mathbf{u}_G^k = \bar{\mathbf{u}}_G^k$ ;
10  fim
11  Impor  $\mathbf{u}_G^k$  como condição de contorno em  $\Gamma$  na mesoescala;
12  Calcular a solução inicial do modelo da mesoescala:  $\mathbf{u}_M$ ;
13  para  $j = 0, 1, \dots$  até  $n_{GL}$  faça
14    Calcular a solução do modelo local:  $\mathbf{u}_{L,j}$ ;
15    Enriquecer o modelo da mesoescala a partir da solução  $\mathbf{u}_{L,j}$ ;
16    Calcular a solução do modelo enriquecido da mesoescala:  $\mathbf{u}_{ME,j}$ ;
17  fim
18  Calcular  $\mathbf{R}^k$  pela Eq. (2.32) utilizando  $\mathbf{u}_G^k$  e  $\mathbf{u}_{ME,n_{GL}}^k$ ;
19  se há convergência então
20    interromper;
21  fim
22   $k = k + 1$ ;
23 fim
```

3.4 Implementação computacional

Os algoritmos apresentados nas Seções 3.1 e 3.3 foram utilizados na elaboração de um programa computacional. O objetivo desse código é realizar o acoplamento não intrusivo entre o modelo global, resolvido pelo MEF, e os modelos da mesoescala e da escala local, resolvido pelo MEF G^{gl} . Esse programa é responsável pelas tarefas associadas ao código “Acoplamento” na Tabela 3.1.

Nesta seção, são apresentadas as principais características do programa. Inicialmente,

Algoritmo 3.5: Versão não intrusiva do método quasi-Newton.

Entrada:

- Iteração k .
- Forças residuais no espaço global $\underline{\mathbf{R}}^k$.
- Incremento da solução prevista $\Delta \bar{\mathbf{u}}_G^k$.

Saída:

- Incremento da solução corrigida $\Delta \mathbf{u}_G^k$.

```

1  $i = 0$ ;
2 enquanto  $i < k$  faça
3   |   Calcular  $(\mathbf{K}_G^{i+1})^{-1} \cdot \underline{\mathbf{R}}^k$  ;           /* Eq. (3.26) */
4   |    $i = i + 1$ ;
5 fim
6  $\Delta \mathbf{u}_G^k = (\mathbf{K}_G^k)^{-1} \cdot \underline{\mathbf{R}}^k$ ;
7
```

são abordadas questões de aspecto geral e, em seguida, são apresentados os principais métodos das classes que constituem o código. O código fonte poderá ser acessado pelo endereço disponível na Seção 5.1.

3.4.1 Características gerais

Como discutido na Seção 1.2, o objetivo do código é realizar a implementação do MEF G^{gl} -GLI. Essa implementação é não intrusiva, de modo que não há alteração nos códigos MEF e MEF G^{gl} envolvidos. Com o seu objetivo bem definido, foram determinadas as características necessárias ou desejadas para o código. Esses requisitos foram motivados pelo objetivo geral deste trabalho, isto é, aproximar *softwares* de pesquisa do contexto industrial e são listados a seguir:

1. *Expansível*: ser facilmente expandido e se tornar compatível com outros programas de análise além daqueles já considerados nesta implementação, sejam eles programas com maior apelo industrial ou acadêmico.
2. *Autonomia*: buscar o mínimo de interações com o usuário. Idealmente, o usuário participa apenas fornecendo os arquivos que contêm os modelos associados a cada uma das três escalas, além das informações sobre a interface. O código deve automatizar todo o processo de acoplamento.
3. *Compartilhável*: é desejável que o código seja facilmente compartilhado e utilizado por outros usuários.

Nesse sentido, o código foi implementado na linguagem Python, em sua versão 3.10. O Python frequentemente figura como a escolha dos desenvolvedores dos programas de análise tanto para suas APIs (*Application Programming Interface*) quanto para a construção de macros e *scripts*. É o caso de alguns programas bastante difundidos na indústria, como o Abaqus, o ANSYS, o SAP 2000 e o Femap *with* NX/Nastran. Por essa razão, muitos usuários desses *softwares* já dominam a linguagem, o que pode facilitar a extensão do código. Além disso, a linguagem possui recursos e bibliotecas nativas que facilitam a construção do código de acoplamento. Por fim, a linguagem é de distribuição gratuita, o que atende à terceira característica desejada.

A implementação foi realizada utilizando o Abaqus para a análise pelo MEF, isto é, para a escala global. Maiores informações sobre o *software* estão disponíveis em Smith (2016). Já para o MEF^{gl}, a implementação do acoplamento foi realizada utilizando a plataforma **INSANE** (Fonseca e Pitangueira, 2007, Pitangueira et al., 2008), já apresentado no Capítulo 1.

Foi adotado o paradigma de programação orientada à objetos (POO). Essa decisão foi tomada com o objetivo de facilitar a implementação do acoplamento para outros pacotes de *software* além dos utilizados neste trabalho, isto é, facilitar a expansão do código.

Por fim, comenta-se sobre a estrutura de dados utilizada para o armazenamento e a manipulação dos vetores de força e de deslocamentos. No escopo do MEF^{gl}-GLI, o acoplamento conduz à manipulação de vetores associados ao modelo global e ao modelo da mesoescala. Nesse sentido, optou-se pela adoção de uma estrutura de dicionários para o armazenamento desses vetores. As entradas do dicionário (*keys*) correspondem aos identificadores dos nós. Os valores do dicionário (*values*) são uma lista (*list*) contendo os valores associados aos graus de liberdade do respectivo nó. Então, por exemplo, se um nó cujo identificador é “Nó 1” está associado à uma força de 1000 unidades na direção x , a sua representação na estrutura de dados será {"Nó 1": [1000.0 0.0 0.0]}. Além de indicar o identificador do nó e as forças associadas a ele, a representação indica, por meio do comprimento da lista, o número de graus de liberdade associados ao nó.

Essa estrutura se mostrou vantajosa por algumas razões. A mais importante delas é que os valores nodais (forças ou deslocamentos) são acessados pelo identificador do nó, de modo que não há preocupação com a posição que esses valores ocupam na estrutura – como aconteceria no caso de listas. Isso facilita a correspondência entre os nós do modelo global e do modelo da mesoescala. Adicionalmente, foram necessárias poucas alterações no código para que ele funcionasse independentemente do número de graus de liberdade em cada nó. Por fim, na linguagem de programação utilizada, a pesquisa de valores em um dicionário (`dict`) é, em geral, mais rápida do que em coleções como listas (`list`) ou tuplas (`tuple`).

3.4.2 Requisitos

O código requer a instalação da linguagem de programação Python na versão 3.10. Também é necessário a instalação da biblioteca `numpy`.

Além disso, devem estar disponíveis os programas responsáveis pelas análises dos modelos computacionais da escala global, mesoescala e escala local. No contexto da implementação deste trabalho, devem estar instalados os programas Abaqus e **INSANE**. O Abaqus foi instalado no computador de forma convencional por meio do instalador disponibilizado pelos desenvolvedores. O **INSANE**, por sua vez, é utilizado por meio de um arquivo executável `.jar`, fornecido no mesmo endereço utilizado para disponibilizar todo o código – ver Seção 5.1.

3.4.3 Classe *coupler*

A classe `coupler` é a principal classe do código elaborado. Ela é responsável pelo acoplamento propriamente dito. É uma classe abstrata, pois alguns de seus métodos podem variar de acordo com os programas de análise envolvidos.

Atributos

A Tabela 3.2 apresenta os principais atributos da classe `coupler`. Foram omitidos alguns atributos criados por conveniência ou por razões práticas não relacionadas ao algoritmo do MEFG^{gl}-GLI.

Em geral, todos os atributos têm sua função e seu uso diretamente obtidos da sua descrição na Tabela 3.2. A exceção é o atributo `results`. Essa variável foi criada para guardar informações de interesse ao longo das iterações. Por exemplo, se há interesse em saber o deslocamento em um dos nós de interface ao longo do processo, essa variável é usada para guardar essas informações. É usada em conjunto com o método `saveResults()`.

Construtor

Um objeto da classe é construído a partir de dois argumentos obrigatórios e dois opcionais. Os dois argumentos obrigatórios são do tipo `model` (ver Seção 3.4.4) e se referem ao modelo global e ao modelo da mesoescala. Não é necessário informar o modelo da escala local. Esse modelo é automaticamente identificado pelo código, o que é possível devido à forma como o MEFG^{gl} está implementado no **INSANE**.

Os argumentos opcionais se referem à convergência. O primeiro é do tipo `double` e representa o valor da tolerância para o critério de convergência – o valor de ϵ na Eq. (3.1). Se nenhum valor for passado, o construtor utilizará o valor padrão $\epsilon = 1,0 \times 10^{-5}$. O segundo parâmetro é um `tuple` de dois elementos que define a técnica de aceleração de convergência a ser utilizada, conforme discutido na Seção 3.3. O primeiro elemento é

Tabela 3.2: Principais atributos da classe `coupler`.

Atributo	Tipo	Descrição
<code>gm</code>	<code>model</code>	Objeto que representa o modelo da escala global. Mais informações na Seção 3.4.4.
<code>mm</code>	<code>model</code>	Objeto que representa o modelo da mesoescala. Mais informações na Seção 3.4.4.
<code>epsilon</code>	<code>double</code>	Valor da tolerância para o resíduo relativo. É o valor de ϵ , definido na Eq. (3.1).
<code>rNormRef</code>	<code>double</code>	Norma euclidiana do vetor de forças residuais calculado na primeira iteração. Consiste em \mathbf{R}^0 , definido na Eq. (3.1).
<code>rtb</code>	<code>accelToolBox</code>	Objeto associado à técnica de aceleração utilizada. Mais detalhes na Seção 3.4.5.
<code>results</code>	<code>list</code>	Utilizado para guardar resultados de interesse a cada iteração. Ao final do processo, é criado um arquivo <code>*.csv</code> contendo as informações coletadas.

um `string` e deve ser escolhida uma das três opções apresentadas a seguir. O segundo elemento do `tuple` é um `double` cujo significado depende do primeiro elemento. As opções para o primeiro parâmetro do `tuple` são:

STATIC Indica o uso da técnica de relaxação estática, discutida na Seção 3.3.1. Nesse caso, o segundo parâmetro é o fator de relaxação a ser utilizado.

DYNAMIC Indica o uso da técnica de relaxação dinâmica, discutida na seção Seção 3.3.2. O segundo parâmetro é o fator de relaxação a ser utilizado nas duas primeiras iterações. Em geral, recomenda-se o uso do valor 1,0.

QNA Indica o uso da técnica de aceleração quasi-Newton, Seção 3.3.3. O segundo parâmetro pode ser utilizado para a definição de um fator de relaxação que será utilizado em conjunto com a técnica quasi-Newton. Essa opção foi implementada, mas será investigada em um trabalho futuro.

Quando executado, o construtor do objeto prepara os arquivos para o acoplamento. É criada uma pasta que receberá uma cópia dos arquivos que contém os modelos. Essa pasta também será utilizada para o armazenamento das informações referentes à todas as iterações, bem como aos resultados obtidos ao final do processo.

Métodos

Apresenta-se a seguir uma descrição dos principais métodos da classe `coupler`. O método `couple` é único necessário para que um usuário realize a análise pelo MEF G^l -GLI. Os demais métodos são de uso interno do programa.

couple() Executa o acoplamento entre os modelos utilizados na construção do objeto `coupler`.

getUcorr() Atualiza o vetor de deslocamentos do modelo global a partir da correção calculada pela técnica de aceleração utilizada.

sendToGlobal() Envia forças nodais para o modelo global, associado ao atributo `gm`.

solveGlobalModel() Soluciona o modelo global, associado ao atributo `gm`.

getGlobalReaction() Recebe do modelo global, referente ao atributo `gm`, as suas reações na interface Γ .

sendToMeso() Envia deslocamentos prescritos para o modelo da mesoescala, associado ao atributo `mm`.

solveMesoModel() Soluciona o modelo da mesoescala, associado ao atributo `mm`.

getMesoReaction() Recebe do modelo da mesoescala, referente ao atributo `mm`, as suas reações na interface Γ .

computeR() calcula as reações residuais referentes na iteração corrente. Esse cálculo é realizado conforme a Eq. (2.32).

checkConvergence() Verifica, ao final de uma iteração, se ocorreu a convergência. A verificação consiste na aplicação da Eq. (3.1), em que ϵ assume o valor do atributo `epsilon` e \mathbf{R}^0 o valor calculado para `rNormRef`.

saveResults() Adiciona um novo elemento à variável `results`, que é do tipo `list`, contendo as informações passadas como argumento.

writeResults() Escreve todos elementos da variável `results` em um arquivo `.csv` na pasta que contém os resultados da análise.

3.4.4 Classe *model*

Esta classe representa um modelo computacional. É uma classe abstrata que é herdada por outras duas classes abstratas: `globalModel` e `mesoModel`, correspondentes aos modelos da escala global e da mesoescala, respectivamente. As classes `globalModel` e `mesoModel` possuem seus métodos próprios, enquanto a classe `model` reúne os métodos e atributos comuns às duas classes que a herdam.

Atributos

A Tabela 3.3 apresenta os atributos da classe `model`. Comenta-se que esses atributos também são os únicos atributos das classes `globalModel` e `localModel`.

Tabela 3.3: Principais atributos da classe `model`.

Atributos	Tipo	Descrição
<code>modelPath</code>	<code>string</code>	Caminho do modelo no sistema de arquivos.
<code>ielem</code>	<code>list</code>	Lista com os identificadores dos elementos adjacentes à interface. Para o modelo global, devem ser listados somente os elementos pertencentes à $\bar{\Omega}_{GC}$. Consultar a Fig. 3.1.
<code>inodes</code>	<code>list</code>	Lista com os identificadores dos nós que se encontram na interface.

Construtor

A construção de uma instância de modelo é realizada pela classe `model`, pois nesse processo não há distinção entre os modelos das escalas global e mesoescala. Devem ser passados três argumentos, cada um correspondendo a um dos três atributos apresentados na Tabela 3.3.

O primeiro argumento deve ser um `string` com o caminho completo do arquivo do modelo. Atenção especial deve ser dada à caracteres especiais, bem como deve-se atentar ao separador de diretórios do sistema (“\”ou “/”). Recomenda-se que o caminho seja inserido entre aspas (“...”) em detrimento ao apóstrofo (‘...’). Isso evita problemas quando existem diretórios com espaços ou caracteres especiais.

A lista dos identificadores dos elementos da interface é o segundo argumento, enquanto o terceiro refere-se aos nós. Atenção deve ser dada ao tipo do identificador. Alguns programas apresentam os identificadores como números inteiros, contudo, no arquivo de saída os identificadores são escritos como `strings`.

Métodos

Lista-se, a seguir, os principais métodos das classes `model`, `globalModel` e `mesoModel`. O conhecimento desses métodos não é necessário para o uso do programa. Ainda assim, tais métodos são necessários para o entendimento do código, bem como para a sua expansão ou adaptação.

- Classe `model`

`run()` Executa a análise do modelo.

`getResults()` Retorna o caminho do arquivo de resultados do modelo.

getU() Retorna o vetor de deslocamentos resultante da solução do modelo.

- Classe `globalModel`

setF() Define um conjunto de forças nodais no modelo global.

getElemForces() Retorna forças nodais nos elementos. No cálculo dos esforços nodais, são considerados apenas os nós e os elementos cujo identificador consta nos atributos `inodes` e `ielem`, respectivamente. Isto é, são considerados apenas os elementos e os nós da interface.

- Classe `mesoModel`

setUp() Define deslocamentos prescritos no modelo da mesoescala.

getReactions() Retorna as reações nos nós do modelo da mesoescala.

3.4.5 Classe *accelToolBox*

Essa é mais uma classe abstrata. Tem o objetivo de permitir a implementação de técnicas de aceleração sem a necessidade de alterações nas demais partes do código. Além disso, essa classe busca facilitar a implementação de técnicas de aceleração reunindo atributos e métodos que seriam comuns entre tais técnicas.

Foi construída uma classe para cada uma das técnicas discutidas na Seção 3.3 e todas elas implementam a `accelToolBox`. A relaxação estática foi implementada pela classe `staticRelax`, enquanto a relaxação dinâmica foi implementada pela classe `dynamicRelax`. Também foi criada a classe `quasiNewtonAccel`, que implementa a técnica de aceleração quasi-Newton. A última classe, como já comentado, não foi testada e será avaliada em um trabalho futuro.

Atributos

A classe `accelToolBox` possui dois atributos. O atributo `type` é do tipo `string` e define o tipo de técnica de aceleração que será utilizada no acoplamento. Na implementação realizada, os valores esperados são aqueles apresentados na Seção 3.4.3, isto é, “STATIC”, “DYNAMIC” e “QNA”. O segundo atributo é marcado pela variável `beta` e representa o fator de relaxação.

A classe `staticRelax` possui o atributo `d_corr`, que armazena a solução corrigida do passo anterior (\mathbf{u}_G^{k-1}). O armazenamento desse vetor é necessário para a obtenção da solução corrigida do passo corrente (\mathbf{u}_G^k), conforme Eq. (3.22).

Na classe `dynamicRelax`, por sua vez, são utilizados dois atributos adicionais. O atributo `d_pred` é uma lista de duas posições. As duas posições são iniciadas vazias. Conforme o código é executado, a primeira posição é ocupada pelo vetor de deslocamentos previstos no passo anterior ($\bar{\mathbf{u}}_G^{k-1}$), enquanto a segunda posição é ocupada pelo vetor de

solução prevista obtido no antepenúltimo passo ($\bar{\mathbf{u}}_G^{k-2}$). O atributo `d_corr` funciona da mesma forma, contudo, para os deslocamentos corrigidos. Esses atributos são utilizados para a atualização do fator de relaxação ω^k , conforme a Eq. (3.23).

Já a classe `quasiNewtonAccel`, por fim, possui quatro atributos adicionais. Foi discutido na Seção 3.3.3 que esse método pode ser implementado de forma recursiva, sendo, para isso, necessário armazenar resultados de todas as iterações que antecederam a iteração corrente. Nesse sentido, as variáveis `r_list`, `du_list` e `kr` são variáveis do tipo `list` nas quais os elementos na posição i armazenam o vetor de forças residuais, o vetor de *incremento* de deslocamentos corrigidos (Linha 6 do Algoritmo 3.5) e o vetor resultante da Eq. (3.26) obtidos na iteração i , respectivamente. A quarta variável, `n`, refere-se ao número de iterações.

A Tabela 3.4 apresenta todos os atributos mencionados, separando-os de acordo com suas respectivas classes.

Construtor

As classes `staticRelax` e `quasiNewtonAccel` são instanciadas passando-se os atributos `type` e `beta` como argumento. Já a classe `dynamicRelax` também requer como argumento o atributo `nodes`. No contexto do MEF G^l -GLI, como comentado, essa lista correspondente ao atributo `inodes` de um objeto da classe `globalModel`.

Métodos

A classe `accelToolBox` possui dois métodos, que são herdados pelas demais. O método `relaxation()` é o único público e é responsável por realizar todo o processo de aceleração, independentemente da técnica implementada. O outro método da classe é o `calculateCorrection()`, que realiza a correção do vetor de deslocamentos previstos. Esses métodos são abstratos na classe `accelToolBox`, de modo que as classes que a implementam realizam a construção dos métodos de forma adequada.

Nas classes `staticRelax` e `dynamicRelax`, são implementados métodos para atualizar, a cada iteração, os atributos `d_corr`, presente tanto em `staticRelax` quanto em `dynamicRelax`, e `d_pred`, que consta somente na classe `dynamicRelax`. A classe `dynamicRelax` também possui o método `updateRelax()` para que o valor do fator de relaxação ω seja atualizado por meio da Eq. (3.25).

A classe `quasiNewtonAccel` não possui métodos além dos herdados da classe-mãe. A seguir, lista-se os métodos existentes na classe `accelToolBox` e nas classes que a implementam.

- Classe `accelToolBox`

`relaxation()` Realiza todo o processo de aceleração, independentemente da técnica utilizada. É o único método público.

Tabela 3.4: Principais atributos das classes `accelToolBox`, `staticRelax`, `dynamicRelax` e `quasiNewtonAccel`.

Atributo	Tipo	Descrição
Classe <code>accelToolBox</code>		
<code>type</code>	<code>string</code>	Tipo de técnica de aceleração adotada.
<code>beta</code>	<code>double</code>	Fator de relaxação.
Classe <code>staticRelax</code>		
<code>d_corr</code>	<code>dict</code>	Vetor de deslocamentos corrigidos da iteração $k - 1$.
Classe <code>dynamicRelax</code>		
<code>nodes</code>	<code>list</code>	Lista com os nós a serem considerados no cálculo da Eq. (3.24). No contexto do MEF gl -GLI, essa lista é populada com os identificadores do nós da interface do modelo global.
<code>d_pred</code>	<code>list</code>	Lista de duas posições. A posição 0 guarda o vetor de deslocamentos previstos na iteração $k - 1$. A posição 1 guarda o vetor de deslocamentos previstos da iteração $k - 2$.
<code>d_corr</code>	<code>list</code>	Lista de duas posições. A posição 0 guarda o vetor de deslocamentos corrigidos da iteração $k - 1$. A posição 2 guarda o vetor de deslocamentos corrigidos da iteração $k - 2$.
Classe <code>quasiNewtonAccel</code>		
<code>r_list</code>	<code>list</code>	Lista que armazena o vetor de forças residuais obtido em cada iteração. O vetor da iteração corrente é adicionado no final da lista.
<code>du_list</code>	<code>list</code>	Lista que armazena o vetor incremento de deslocamento de cada iteração. É calculado conforme a Linha 6 do Algoritmo 3.5.
<code>kr</code>	<code>list</code>	Lista que armazena o vetor resultante da Eq. (3.26) a cada iteração.
<code>n</code>	<code>int</code>	Número da iteração corrente. Necessário para a definição o conjunto de valores a serem assumidos por i no Algoritmo 3.5.

calculateCorrection() Realiza a correção do vetor de deslocamentos previstos.

- Classe `staticRelax`

updateCorr() Atualiza o vetor de deslocamentos corrigidos (\mathbf{u}_G) armazenado no atributo `d_corr`. Esse vetor será utilizado na Eq. (3.22) para a correção da solução da próxima iteração.

- Classe `dynamicRelax`

updateCorr() Atualiza o vetor de deslocamentos corrigidos (\mathbf{u}_G) armazenado no atributo `d_corr`. Esse vetor será utilizado na Eq. (3.23) para a correção da solução da próxima iteração.

updatePred() Atualiza o vetor de deslocamentos previstos ($\bar{\mathbf{u}}_G$) armazenado no atributo `d_pred`. Esse vetor será utilizado na Eq. (3.23) para a correção da solução da próxima iteração.

updateRelax() Atualiza o valor do fator de relaxação. É uma aplicação direta da Eq. (3.25).

3.4.6 Classe *linAlg*

Foi comentado que os vetores de forças e de deslocamentos são armazenados no código por meio de dicionários. A classe `linAlg` foi, então, concebida para permitir a realização de operações vetoriais simples a partir dessa estrutura de dados. Ela fornece métodos que realizam operações básicas como a combinação linear entre dois vetores, por exemplo. É uma classe que não pode ser instanciada. Não há, portanto, atributos nem método construtor. O seu objetivo é fornecer métodos auxiliares para as demais classes. Tais métodos são comentados a seguir.

getVectorFromDict() Retorna todos os valores (`values`) de um dicionário como uma lista (`list`). A lista é construída pela ordem das entradas (`keys`) do dicionário.

getNormFromDict() Retorna a norma euclidiana do vetor formado pelos valores (`values`) de um dicionário.

getInnerFromDict() Retorna o produto interno entre dois vetores representados por dicionários. Os vetores são obtidos a partir dos valores (`values`) dos dicionários.

getLinearCombFromDict() Retorna a combinação linear entre dois vetores representados por dicionários. Os vetores são obtidos a partir dos valores (`values`) dos dicionários. Cada vetor pode ser multiplicado por um escalar.

getLinearComb() Retorna a combinação linear entre dois vetores representados por listas. Cada vetor pode ser multiplicado por um escalar.

Capítulo 4

Simulações numéricas

Neste capítulo são apresentados os resultados de simulações numéricas realizadas com o objetivo de validar a implementação do MEF G^{gl} -GLI e das técnicas de aceleração de convergência apresentadas na Seção 3.3. Após a validação da implementação, foram realizadas investigações acerca dos principais parâmetros do MEF G^{gl} -GLI.

Os resultados foram comparados com modelos de referência. Os modelos de referência foram construídos explorando-se o conceito apresentado na Seção 2.4 – ver Fig. 2.9. Assim, as soluções dos problemas apresentados neste capítulo também foram obtidas por modelos de elementos finitos convencional com malha refinada, pelo menos, no domínio local.

4.1 Solução pelo MEF-GLI

Inicialmente, buscou-se aplicar o não intrusivo proposto por Whitcomb (1991) na solução de um problema simples. Também buscou-se verificar a influência do uso do MEF-GLI em diferentes cenários de rigidez do domínio local, dado o exposto na Seção 3.2.2. O Problema 1 é apresentado na Fig. 4.1. Optou-se por um problema unidimensional, tão simples quanto possível. A barra, que possui 9 m de comprimento, está fixa em uma de suas extremidades. Na outra extremidade atua uma força axial de 150 kN. A seção da barra possui área A e é constante ao longo de todo o comprimento.

O problema foi discretizado por meio de elementos de barra de dois nós. Cada nó possui um grau de liberdade na direção axial do elemento. O modelo global foi construído com três elementos. O domínio local foi definido pelo trecho da barra representado pelo elemento central do modelo global. No modelo local, essa região foi discretizada utilizando o mesmo tipo de elemento, porém, com refinamento. Foram utilizados elementos quatro vezes menores. A Fig. 4.1 apresenta os modelos descritos.

Não foram consideradas não linearidades físicas ou geométricas. Desse modo, a ocorrência de efeitos locais foi introduzida por meio da alteração do módulo de elasticidade associado a alguns dos elementos do modelo local. Todos os elementos do modelo global

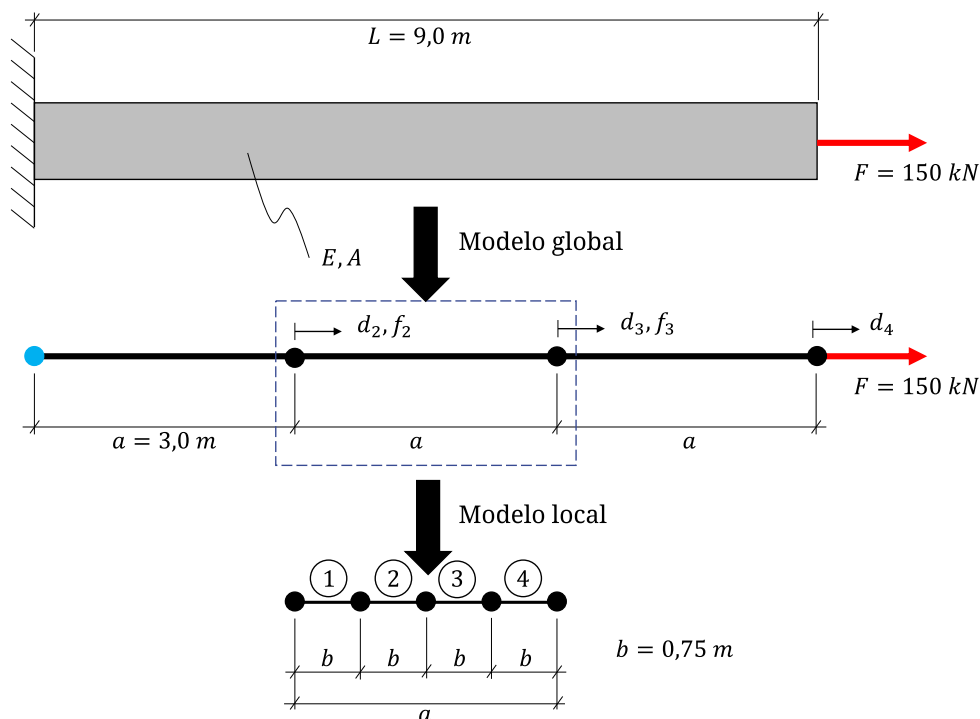


Figura 4.1: Problema 1 – Barra submetida a força concentrada em uma das extremidades.

possuem módulo de elasticidade E . No modelo local, por sua vez, o mesmo módulo de elasticidade E foi utilizado nos elementos de extremidade (na Fig. 4.1, elementos 1 e 4), enquanto os elementos centrais (2 e 3) foram considerados com módulo de elasticidade $E^* = \lambda E$, em que $\lambda \in \mathbb{R}^+ \mid \lambda \neq \{0, 1\}$ é um valor predefinido. Foram considerados dois casos. No Caso 1, assumiu-se $\lambda = 0,5$. Este caso representa o cenário no qual $\bar{\Omega}_L$ é discretizado de forma mais flexível no modelo local em comparação com o modelo global. O Caso 2 foi definido com $\lambda = 2,0$ e, assim, representa o cenário em que a representação do domínio local é mais rígida no modelo local.

Uma vez que o objetivo é avaliar o comportamento do procedimento de acoplamento pelo GLI, os dois cenários foram resolvidos pelo MEF-GLI por meio de solução *manual*, no sentido de que não foram utilizados programas computacionais que implementam o MEF. A solução foi obtida com o auxílio de planilha eletrônica construída particularmente para o Problema 1. A implementação computacional do GLI, bem como do MEFGLI-GLI, será abordada nas próximas seções. Foram realizadas as 10 primeiras iterações de cada cenário.

Tomou-se para avaliação dos resultados a força residual no nó 3 do modelo global. Os resultados obtidos estão apresentados na Fig. 4.2. Ainda que nos dois casos a força residual apresente tendência a valores cada vez mais próximos de zero, observa-se que no Caso 1 isso se dá forma monotônica, o que não ocorre no Caso 2.

A Fig. 4.3 apresenta que a taxa de convergência do MEF-GLI se dá em escala logarítmica em relação ao módulo da força residual para o Caso 1. O comportamento obtido está de acordo com o indicado por outros autores (Duval et al., 2014, Li et al., 2021).

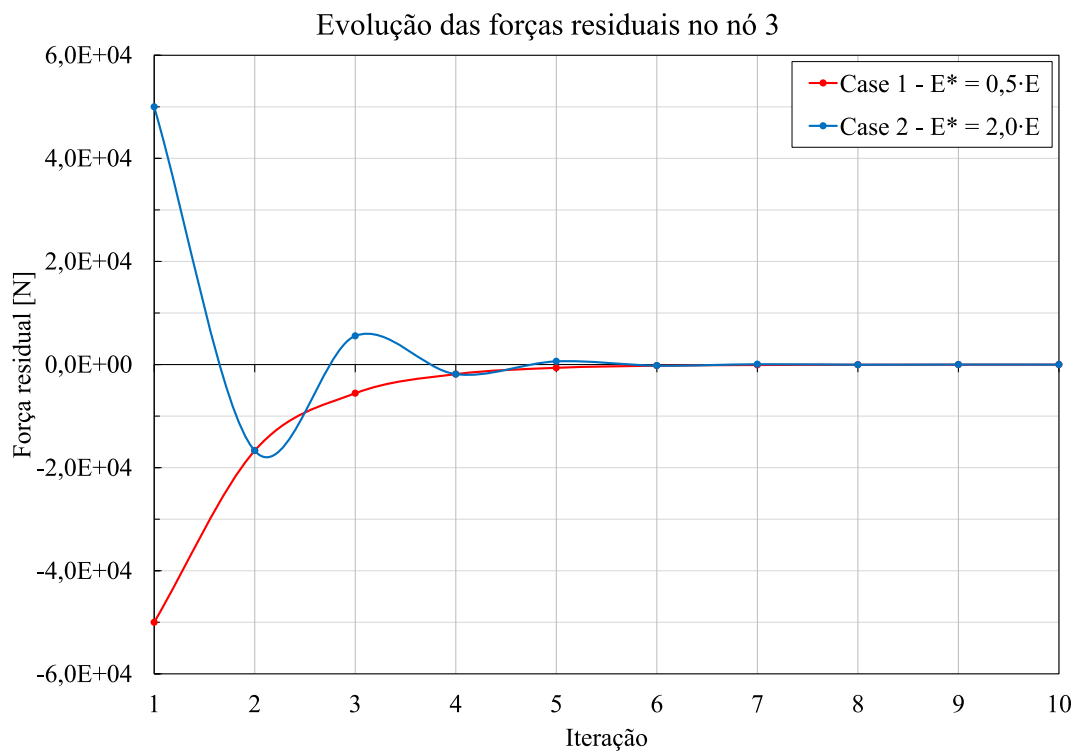


Figura 4.2: Forças residuais no MEF-GLI — Ω_L mais rígido (azul) e menos rígido (vermelho) no modelo local.

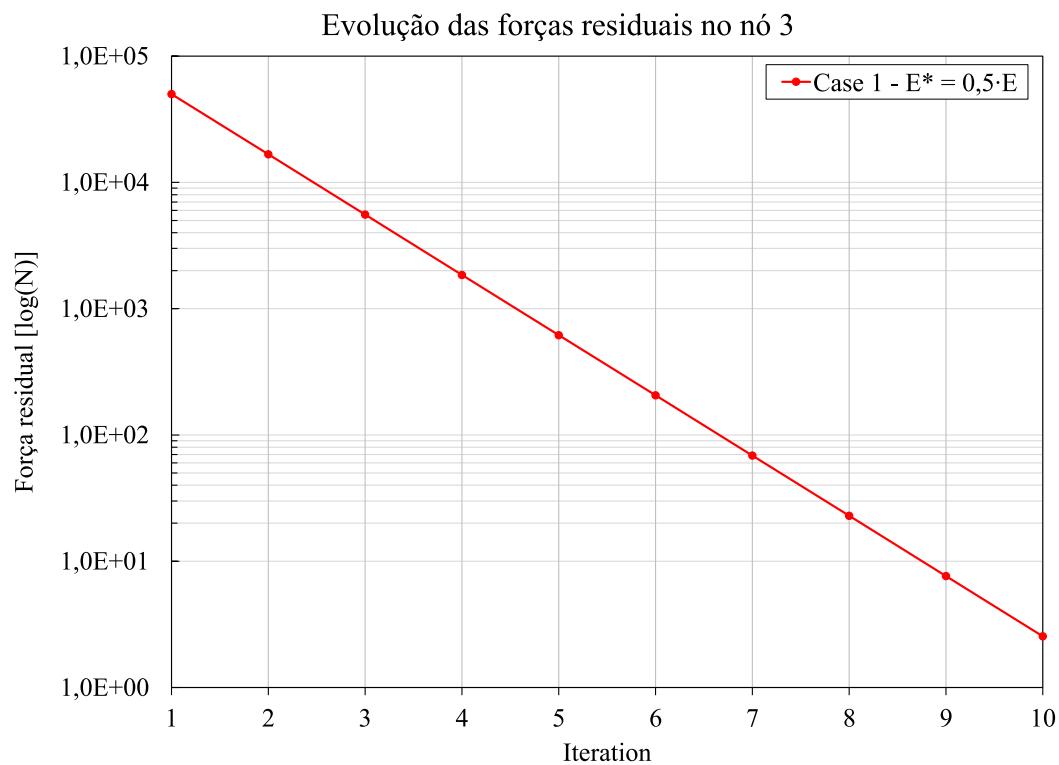


Figura 4.3: Taxa de convergência típica do MEF-GLI para problemas lineares elásticos.

Tabela 4.1: Comparação de resultados entre o MEF e o MEF-GLI.

Coord. [m]	Deslocamento [m]					
	Caso 1			Caso 2		
	MEF	MEF-GLI	dif. (%)	MEF	MEF-GLI	dif. (%)
0,00	0,000E+00	0,000E+00	—	0,000E+00	0,000E+00	—
3,00	4,500E−01	4,500E−01	0,00000%	4,500E−01	4,500E−01	0,00000%
3,75	5,625E−01	5,625E−01	0,00003%	5,625E−01	5,625E−01	0,00004%
4,50	7,875E−01	7,875E−01	0,00007%	6,188E−01	6,187E−01	0,00005%
5,25	1,013E+00	1,013E+00	0,00009%	6,750E−01	6,750E−01	0,00006%
6,00	1,125E+00	1,125E+00	0,00009%	7,875E−01	7,875E−01	0,00008%
9,00	1,575E+00	1,575E+00	0,00007%	1,238E+00	1,237E+00	0,00005%

Quanto à qualidade dos resultados, a Tabela 4.1 compara os deslocamentos obtidos pelo MEF convencional e pelo MEF-GLI nos nós dos modelos. Tanto no Caso 1 quanto no Caso 2, a aplicação do GLI apresentou resultados equivalentes, isto é, não há diferença significativa em relação à solução MEF convencional. A Fig. 4.4 apresenta a solução de referência para o Caso 2. É dado destaque ao que pode ser chamado de “montagem” do modelo de referência, conforme discutido na Seção 2.4, e sua solução foi obtida pelo *software* Femap *with* NX/Nastran. Ao se utilizar o MEF-GLI, a melhor solução possível corresponde àquela obtida por um modelo computacional que seria composto pela discretização do modelo global em Ω_{GC} e pela discretização do modelo local em Ω_L . Essa é a solução apresentada na Fig. 4.4 para o Caso 2.

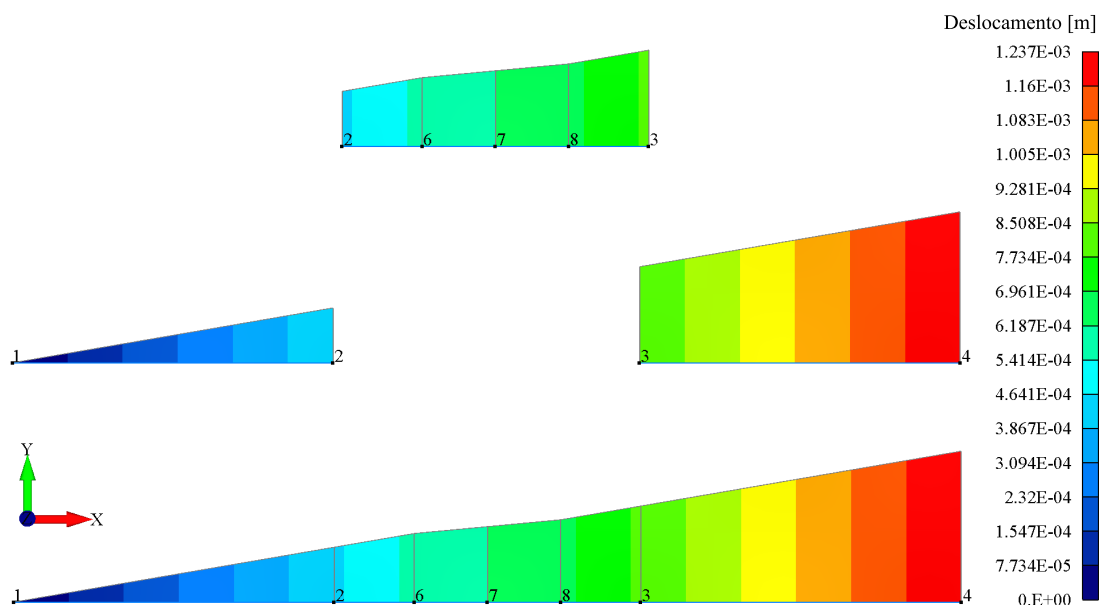


Figura 4.4: Solução de referência para o Caso 2 do Problema 1.

4.2 Validação do MEF G^{gl} -GLI no INSANE

Após a avaliação do MEF-GLI por meio da solução manual do Problema 1, seguiu-se para a implementação do MEF G^{gl} -GLI. Nesse primeiro momento, as três escalas de análise são resolvidas pelo **INSANE**.

Essa é, portanto, uma etapa intermediária, tendo em vista que o objetivo do trabalho é implementar o MEF G^{gl} -GLI de forma que a escala global seja solucionada por um programa com apelo industrial. Esse passo in abordagem foi adotada por razões práticas. Conforme a Tabela 3.1, o escopo referente à implementação do acoplamento pode ser dividido em três fases, a saber: troca de informações com o modelo da escala global (MEF), troca de informações com o modelo da mesoescala (MEF G^{gl}) e processamento do acoplamento propriamente dito. Ao se utilizar o mesmo *software* para a solução de todas as escalas, a implementação é facilitada pois, inicialmente, o acoplamento deve ser capaz de trocar informações com apenas um *software*. Como, nesta etapa, o objetivo é validar a implementação do MEF G^{gl} -GLI, o uso do **INSANE** para a solução de todos os modelos facilita o processo. Ressalta-se que a implementação foi posteriormente estendida para que a solução do modelo global fosse realizada pelo Abaqus, o que será tema da próxima seção.

A validação da implementação foi realizada por meio da solução do Problema 2, apresentado na Fig. 4.5, que consiste em uma chapa sob estado plano de tensões (EPT). A chapa possui um orifício quadrado em sua região central. O carregamento é de tração e é autoequilibrado. A figura também apresenta as dimensões e os outros dados do problema, todos em unidades consistentes. Não há não linearidade física ou geométrica, de modo que o orifício é a única fonte de efeitos localizados. O material é homogêneo e isotrópico, com módulo de elasticidade $E = 2,0 \times 10^{11}$ u.c. e coeficiente de Poisson $\nu = 0,30$.

Conforme o MEF G^{gl} -GLI, foram construídos 3 modelos estruturais computacionais. Todos eles utilizam elementos finitos do tipo Q4 com ordem de integração 4x4. As malhas dos modelos da escala global e da mesoescala possuem elementos com dimensão igual à do orifício, isto é, os elementos possuem lado com dimensão $l = 5,0$. Não há, contudo, coincidência geométrica entre o orifício e o centro de qualquer elemento – o orifício está posicionado entre quatro elementos, no centro da chapa. A malha do modelo local, única que representa o orifício, foi construída com elementos de dimensão $l = 2,5$. A Fig. 4.5 também apresenta as malhas utilizadas. Em cada modelo, a região em tom mais claro indica o domínio da escala inferior – o vermelho mais claro no modelo da escala global indica o domínio da mesoescala, por exemplo.

Os resultados obtidos pela implementação do MEF G^{gl} -GLI utilizando exclusivamente o **INSANE** são apresentados na Fig. 4.6 em termos da componente de deslocamentos na direção y (u_y). Inicialmente, observa-se que a solução reflete a simetria inerente ao problema. No contexto deste trabalho, essa observação é importante, pois é um forte

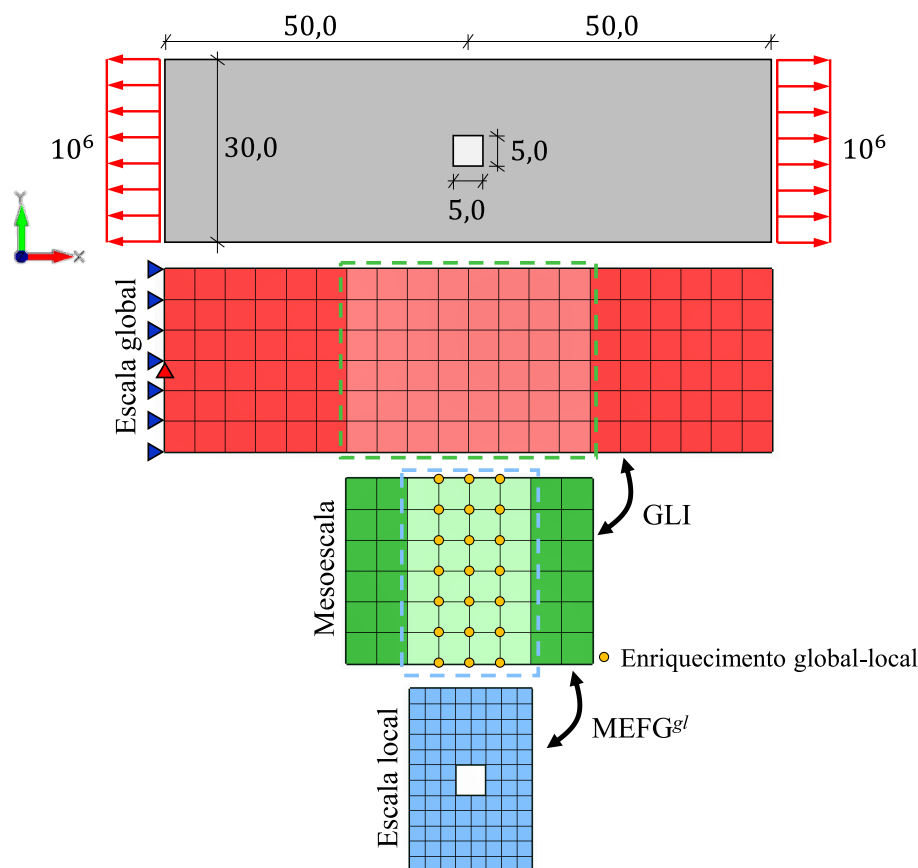


Figura 4.5: Problema 2 – Chapa com orifício central sob tração pura. Unidades consistentes.

indicativo em relação à implementação da transferência de forças e de carregamentos entre os modelos.

Observa-se que, ainda que o modelo global não considere os efeitos locais, a sua resposta final em Ω_{GC} é corrigida. Há leve incremento no campo de deslocamentos ao se aproximar da região central da chapa. O modelo da mesoescala, modificado pela introdução de funções de enriquecimento global-local, representa de forma satisfatória a perda de rigidez devido ao orifício. Os maiores deslocamentos verticais, em módulo, ocorrem nas fibras superior e inferior da seção central e o valor associado é $2,781 \times 10^{-5}$ u.c.. Essa solução foi obtida a partir dos 690 graus de liberdade dos modelos referentes às três escalas.

Foram realizadas análises com o objetivo de avaliar a qualidade do resultado obtido. Nesse sentido, o MEF foi utilizado em sua formulação convencional por meio do *software* Femap *with* NX Nastran. Novamente, foram utilizados elementos finitos do tipo Q4. O Modelo A foi construído com malha quadrilateral de lado $l = 2,5$, o que resultou em 1.064 graus de liberdade. O segundo modelo, dito Modelo B, teve sua malha definida com $l = 0,625$ e, assim, possui 15.680 graus de liberdade.

As soluções obtidas pelo Modelo A e pelo Modelo B são apresentadas nas Figs. 4.7 e 4.8, respectivamente, em termos dos deslocamentos verticais. Para o Modelo A, observa-se

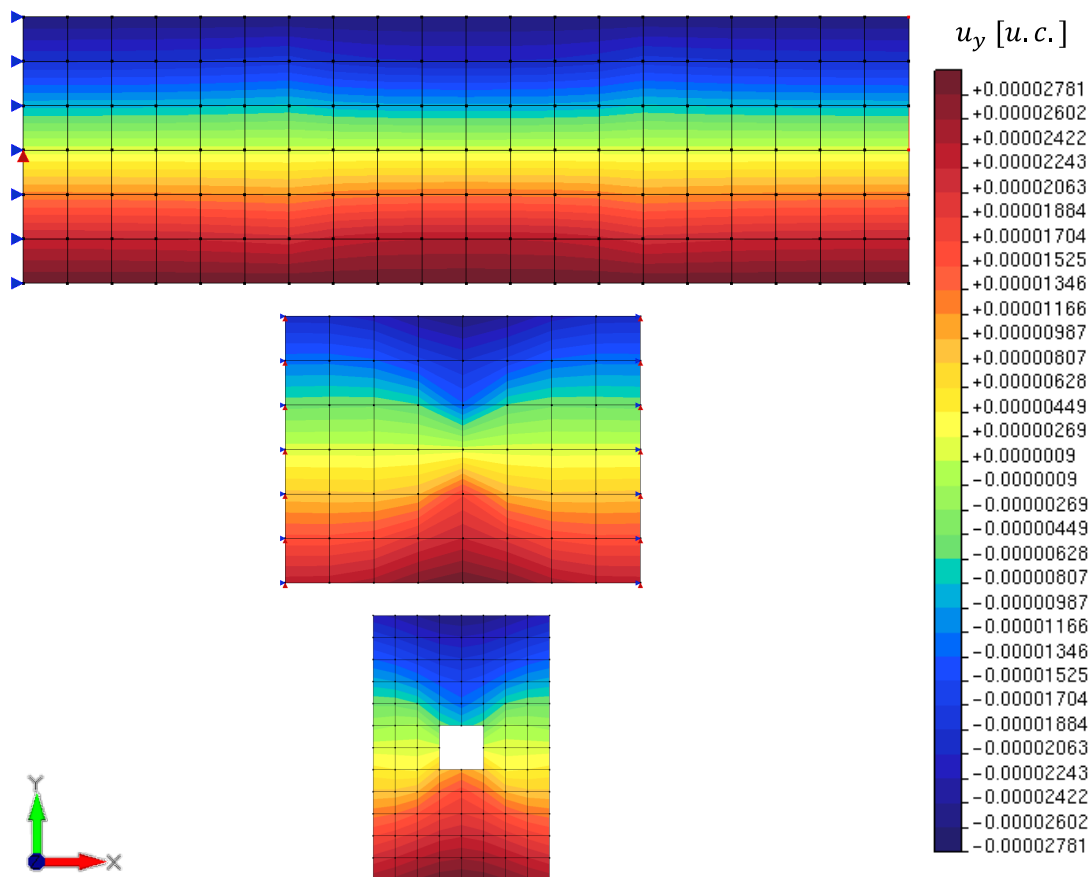


Figura 4.6: Deslocamentos na direção y (u_y) obtidos pelo MEF G^{gl} -GLI.

que o campo de deslocamentos possui fortes semelhanças com o obtido pelo MEF G^{gl} -GLI. Além da simetria em relação aos planos horizontal e vertical, observa-se que a solução do Modelo A indica deslocamentos com mesmo valor nas duas bordas verticais, leve redução na magnitude dos deslocamentos ao se aproximar do terço central da chapa e forte aumento na magnitude dos deslocamentos na região do orifício. O comportamento descrito está graficamente representado na Fig. 4.9, que será retomada após os comentários sobre o Modelo B. Ainda em relação ao Modelo A, cabe registrar que o valor máximo, em módulo, obtido para a componente de deslocamento na direção y é igual a $2,769 \times 10^{-5}$ e ocorre nas fibras de extremidade da seção central.

Na solução obtida por meio do Modelo B, a componente de deslocamento vertical possui o valor máximo de $2,831 \times 10^{-5}$, novamente nas fibras de extremidade da seção central. Este modelo é mais fidedigno do que os anteriores devido ao maior número de graus de liberdade. Já era, portanto, esperado que o deslocamento obtido por esse modelo fosse de maior magnitude em comparação com os demais e a resposta do Modelo B será admitida como valor de referência.

Ainda que o Problema 2 seja simples, a introdução do orifício resulta em significativa alteração no campo de deslocamentos. A Fig. 4.9 apresenta a componente de deslocamentos na direção y obtidos ao longo da borda inferior da chapa. Observa-se que o resultado

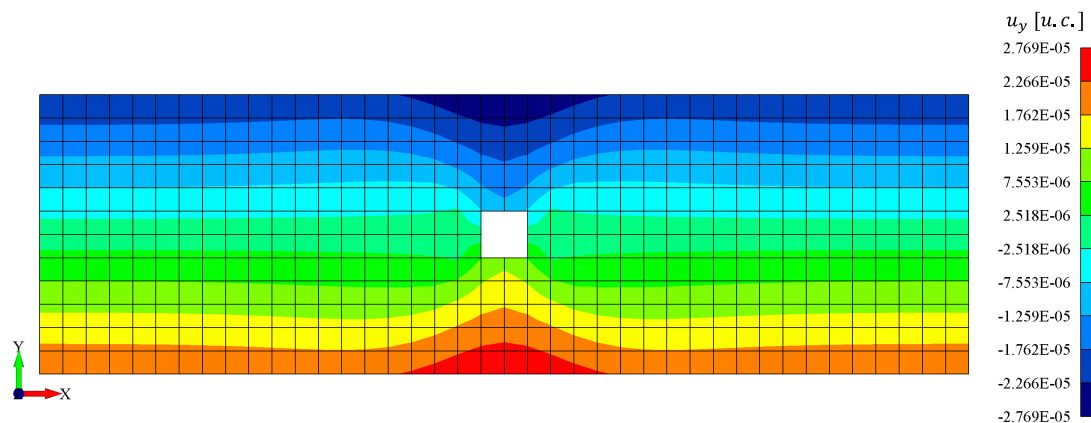


Figura 4.7: Deslocamentos na direção y (u_y) obtidos pelo Modelo A (1.064 graus de liberdade).

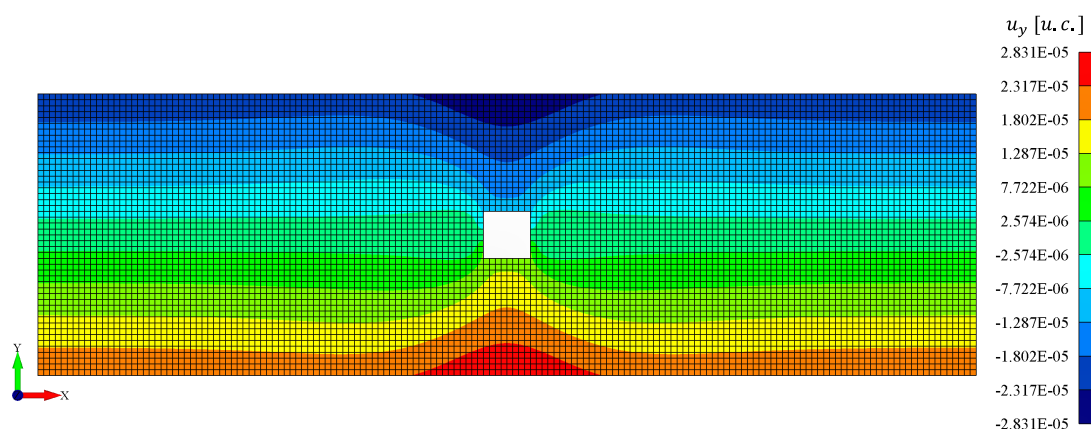


Figura 4.8: Deslocamentos na direção y (u_y) obtidos pelo Modelo B (15.680 graus de liberdade).

do $\text{MEFG}^{gl}\text{-GLI}$ está entre os obtidos pelo Modelo A e pelo Modelo B. A solução pelo $\text{MEFG}^{gl}\text{-GLI}$ é mais interessante do que a obtida pelo Modelo A, pois utiliza menos graus de liberdade (690 contra 1.064) e a aproximação obtida está mais próxima do valor de referência ($2,781 \times 10^{-5}$ contra $2,769 \times 10^{-5}$).

Outro aspecto importante do gráfico da Fig. 4.9 diz respeito aos resultados obtidos na interface Γ entre a escala global e a mesoescala. Neste problema, existem duas regiões de interface posicionadas, cada uma, à 30 u.c. (unidades consistentes) da borda vertical mais próxima. No gráfico, portanto, há interface nas posições 30 u.c. e 70 u.c.. A avaliação do gráfico indica que, em termos de deslocamentos, a solução pelo $\text{MEFG}^{gl}\text{-GLI}$ não introduziu resultados espúrios na região.

4.3 Validação do $\text{MEFG}^{gl}\text{-GLI}$ no Abaqus-INSANE

Apresenta-se, finalmente, os resultados obtidos a partir da implementação referente à proposta deste trabalho. Nesta seção, os resultados correspondem à implementação do $\text{MEFG}^{gl}\text{-GLI}$ utilizando o Abaqus como ferramenta computacional para a solução do modelo global por meio do MEF e o **INSANE** para a solução dos modelos da mesoescala

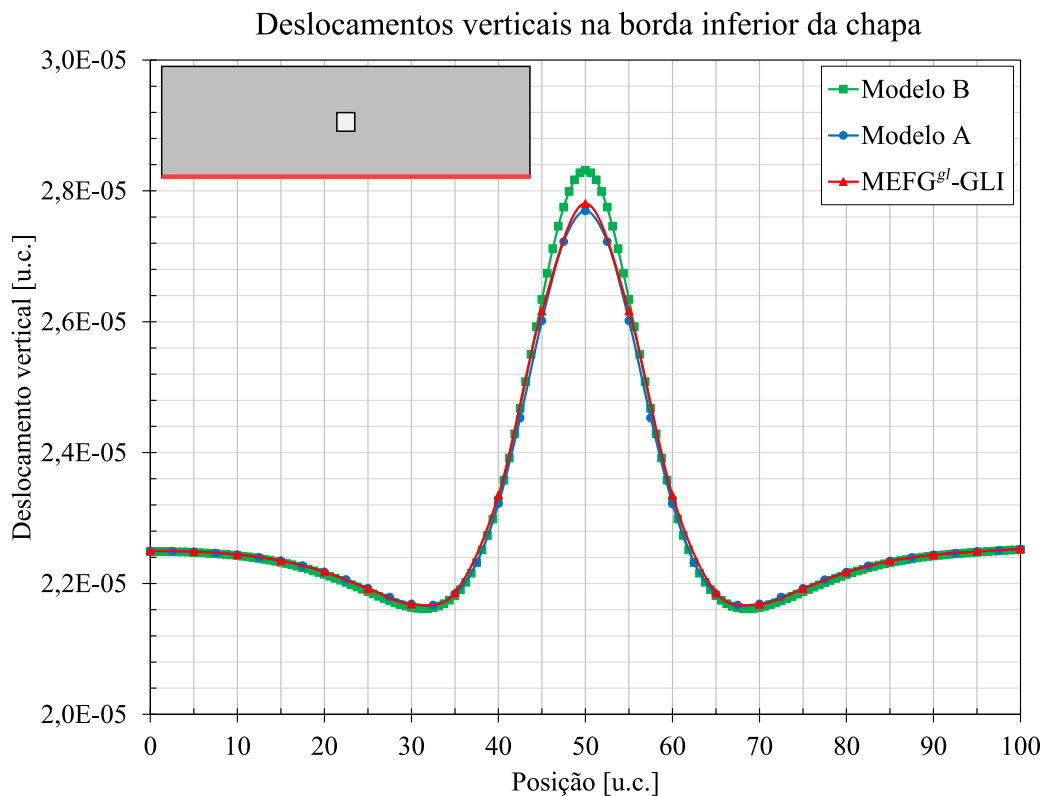


Figura 4.9: Comparação dos deslocamentos u_y na borda inferior da chapa obtidos pelo MEF G^{gl} -GLI e pelos Modelos A e B.

e da escala local pelo MEF G^{gl} .

Inicialmente, investigou-se os parâmetros inerentes ao MEF G^{gl} -GLI. Nesse sentido, foram feitas simulações com o objetivo de avaliar a influência do valor da tolerância na resposta final obtida – o valor de ϵ na Eq. (3.1). Em seguida, investigação similar foi realizada em relação ao número de iterações global-local na solução da mesoescala, isto é, iterações no MEF G^{gl} – o valor de n_{GL} nos Algoritmos 3.1 a 3.4. A abrangência da mesoescala também teve sua influência avaliada. Por fim, foram avaliadas algumas das técnicas de aceleração da convergência discutidas na Seção 3.3 por meio de diferentes problemas.

Em geral, as investigações foram realizadas a partir do Problema 2, já apresentado na Fig. 4.5. Algumas modificações foram realizadas de acordo com o interesse de cada situação de investigação. Essas modificações serão apresentadas sempre que necessário. Ressalta-se que o problema não contém não linearidades físicas ou de geometria e que os resultados e comentários apresentados nesta seção referem-se a tal classe de problemas.

4.3.1 Influência da tolerância

O primeiro parâmetro do MEF G^{gl} -GLI a ser avaliado foi a tolerância em relação ao resíduo relativo, definida pela relação entre as normas euclidianas do vetor de forças residuais da iteração k , \mathbf{R}^k , e o vetor de forças residuais da primeira iteração, \mathbf{R}^0 . Essa

relação foi apresentada na Eq. (3.1), na qual a tolerância é representada por ϵ .

Li, O'Hara e Duarte (2021) indicam que valores de tolerância da ordem de 10^{-5} conduzem a resultados suficientemente precisos. Contudo, o trabalho mencionado não apresentou um estudo em relação a isso. Whitcomb (1991), por sua vez, apresentou uma investigação em relação ao critério de convergência. Em seu trabalho, a convergência era atingida quando todos os elementos que compõem o vetor \mathbf{R} apresentavam valor absoluto inferior a ϵ . Desse modo, o critério de convergência não era relativo. Além disso, Whitcomb avaliou a convergência do MEF-GLI, enquanto aqui se investiga o MEFG^{gl}-GLI.

Nesse sentido, com o objetivo de investigar a influência do valor atribuído a ϵ nos resultados, um problema foi resolvido pelo MEFG^{gl}-GLI adotando-se cinco valores distintos de convergência: 10^{-2} , 10^{-3} , 10^{-5} , 10^{-7} e 10^{-10} . O Problema 2, apresentado na seção Seção 4.2, foi novamente considerado. A única observação é que agora o modelo da escala global foi solucionado pelo Abaqus. As malhas apresentadas na Fig. 4.5 são aqui novamente utilizadas. As soluções apresentadas nesta seção foram obtidas utilizando-se apenas um ciclo MEFG^{gl} por iteração do GLI, isto é, $n_{GL} = 1$. Esse parâmetro será estudado na próxima seção.

Os resultados foram avaliados em termos das componentes de deformação. A Fig. 4.10 apresenta a resposta obtida pelo MEFG^{gl}-GLI utilizando o menor e o maior valor de tolerância utilizados, respectivamente 10^{-10} e 10^{-2} . A Fig. 4.11 apresenta os resultados obtidos pela formulação convencional do MEF por meio dos Modelos A e B, definidos na Seção 4.2. Nas figuras, os resultados visualizados compreendem somente o domínio da mesoescala.

Sabe-se que o Modelo B fornece solução mais precisa do que o Modelo A devido ao refinamento h . Nesse sentido, observa-se que, independentemente da tolerância utilizada, em termos da componente de deformação ϵ_{yy} , os resultados obtidos pelo MEFG^{gl}-GLI são coerentes. Recordar-se que o Modelo A possui 1064 graus de liberdade, enquanto os três modelos utilizados no MEFG^{gl}-GLI somam 690 graus de liberdade.

A comparação dos dois resultados apresentados na Fig. 4.10 indica que não houve significativa mudança no campo de deformações obtido ao se variar o valor da tolerância. Há, contudo, certa diferença nos resultados e, nesse sentido, foram definidos dois pontos para uma comparação mais precisa. Outra razão para a definição de pontos para a comparação de resultados reside no fato de o modelo da mesoescala não contemplar o orifício. Os pontos selecionados estão indicados na Fig. 4.12. Foi estabelecido como requisito para a definição dos pontos que todos os modelos apresentassem um nó nessa posição.

O Ponto 1 foi selecionado, pois a solução do Modelo B indica que ocorre aumento de deformação nessa região. Esse aumento foi capturado pela solução MEF do Modelo A, menos refinada (ver Fig. 4.11). O objetivo é avaliar se a solução pelo MEFG^{gl}-GLI, ainda menos refinada, também seria capaz de aproximar bem o resultado, bem como avaliar se

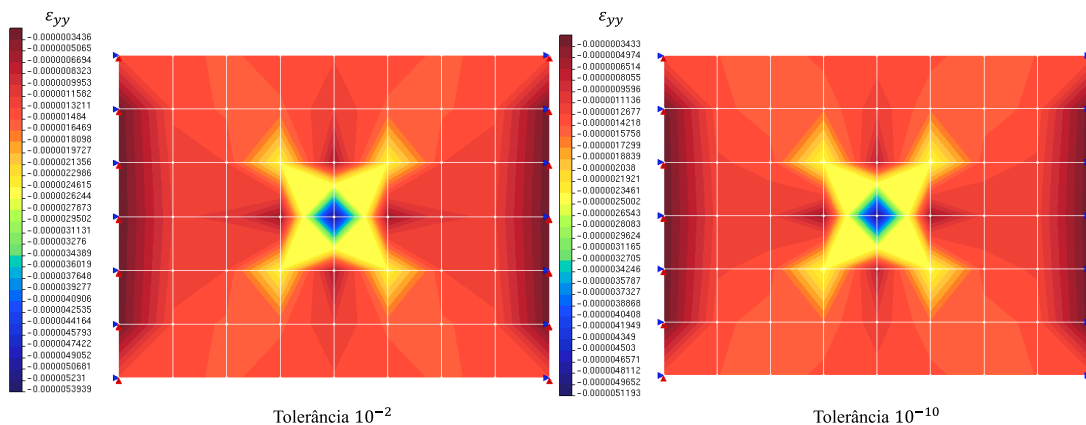


Figura 4.10: Componente de deformação ε_{yy} na mesoescala. Resultados obtidos pelo MEF G^{gl} -GLI.

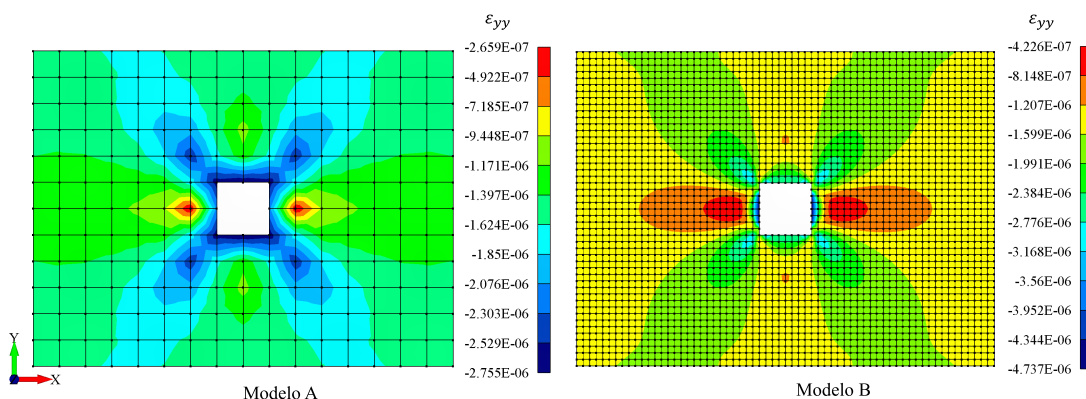


Figura 4.11: Componente de deformação ε_{yy} na mesoescala. Resultados obtidos pelo MEF.

a tolerância implica em alguma alteração de resultado na região.

Já o Ponto 2, por sua vez, foi selecionado por dois motivos. As Figs. 4.10 e 4.11 indicam que nessa região ocorre o menor valor de ε_{yy} . Ainda, espera-se que a componente de deformação ε_{xy} seja nula ou numericamente insignificante nesse ponto.

A Tabela 4.2 compara os resultados obtidos por cada uma das soluções realizadas em termos das componentes de deformação nos Pontos 1 e 2. Os resultados obtidos pelo MEF G^{gl} -GLI indicam que não há variação de resultados para valores de tolerância superiores a 10^{-5} . Mesmo quando adotado $\epsilon = 10^{-2}$, os resultados obtidos estão distantes cerca de 1% dos obtidos com a menor tolerância considerada ($\epsilon = 10^{-10}$). Essa conclusão também pode ser obtida graficamente pela Fig. 4.13.

Observa-se que há congruência entre os resultados obtidos pelas diferentes soluções. Avaliando-se a componente de deformação na direção x , os resultados obtidos pelo MEF G^{gl} -GLI apresentam diferença média de 3,50% e 1,20% em relação aos valores obtidos pelo Modelo B no Ponto 1 e no Ponto 2, respectivamente. O Modelo A apresenta diferença de 1,38% no Ponto 1, e de -6,40% no Ponto 2, também em relação ao Modelo B. Quanto à componente de deformação na direção y , o MEF G^{gl} -GLI apresentou resultados com variação de 9,30% e 10,34% em relação ao Modelo B nos pontos 1 e 2. O Modelo A esteve

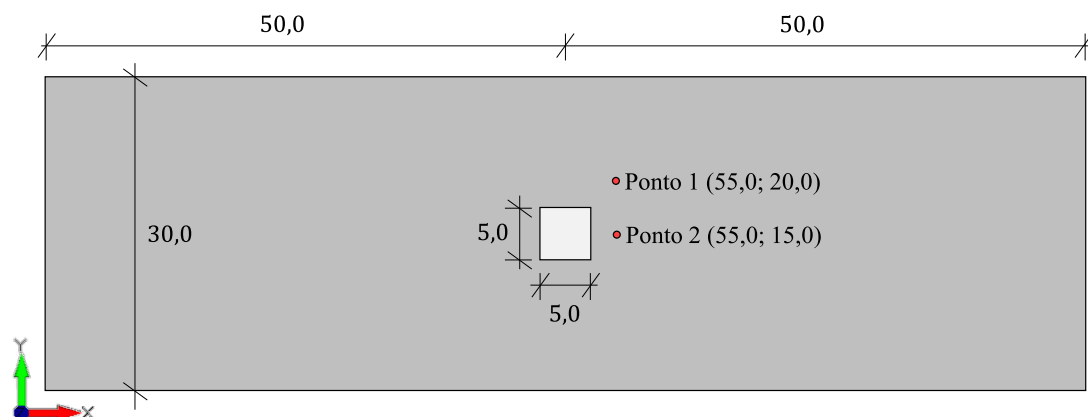


Figura 4.12: Definição dos pontos 1 e 2 para comparação de resultados.

um pouco mais próximo dos resultados do Modelo B, com variações de 2,37% e $-6,52\%$, respectivamente. Finalmente, a avaliação das deformações no plano xy indica boa precisão do MEF gl -GLI. No Ponto 1, os resultados foram 2,54% superiores aos obtidos pelo Modelo B. O Modelo A, por sua vez, apresentou nesse mesmo ponto uma deformação 18,21% superior em relação ao Modelo B. No Ponto 2, todos os modelos apresentaram $\epsilon_{xy} = 0$, conforme esperado.

Tabela 4.2: Comparação de resultados do MEF gl -GLI para diferentes valores de tolerância (ϵ).

Tolerância (ϵ)	Componentes de deformação (10^{-6})					
	Ponto 1			Ponto 2		
	ϵ_{xx}	ϵ_{yy}	ϵ_{xy}	ϵ_{xx}	ϵ_{yy}	ϵ_{xy}
10^{-2}	6,2743	-2,6254	-0,8795	1,6734	-0,6708	0,0000
10^{-3}	6,2757	-2,6260	-0,8797	1,6738	-0,6710	0,0000
10^{-5}	6,2757	-2,6260	-0,8798	1,6738	-0,6710	0,0000
10^{-7}	6,2757	-2,6260	-0,8798	1,6738	-0,6710	0,0000
10^{-10}	6,2757	-2,6260	-0,8798	1,6738	-0,6710	0,0000
Modelo A	6,1472	-2,4597	-1,0142	1,2127	-0,2659	0,0000
Modelo B	6,0636	-2,4028	-0,8580	1,6008	-0,4226	0,0000

Os resultados apresentados até então indicam que valores de tolerância superiores a 10^{-5} não impactam a qualidade das respostas obtidas. A tolerância adotada, contudo, também deve ser avaliada em relação ao custo computacional para a obtenção da solução. No escopo do MEF gl -GLI, a avaliação do custo computacional foi realizada em termos do número de iterações necessárias para a convergência e do tempo de execução. Em vez de avaliar o tempo de execução por meio de valores absolutos, optou-se por uma comparação relativa em relação ao tempo gasto com a maior tolerância considerada. Portanto, o tempo de execução da solução obtida com $\epsilon = 10^{-2}$ é o valor de referência e vale 1,0.

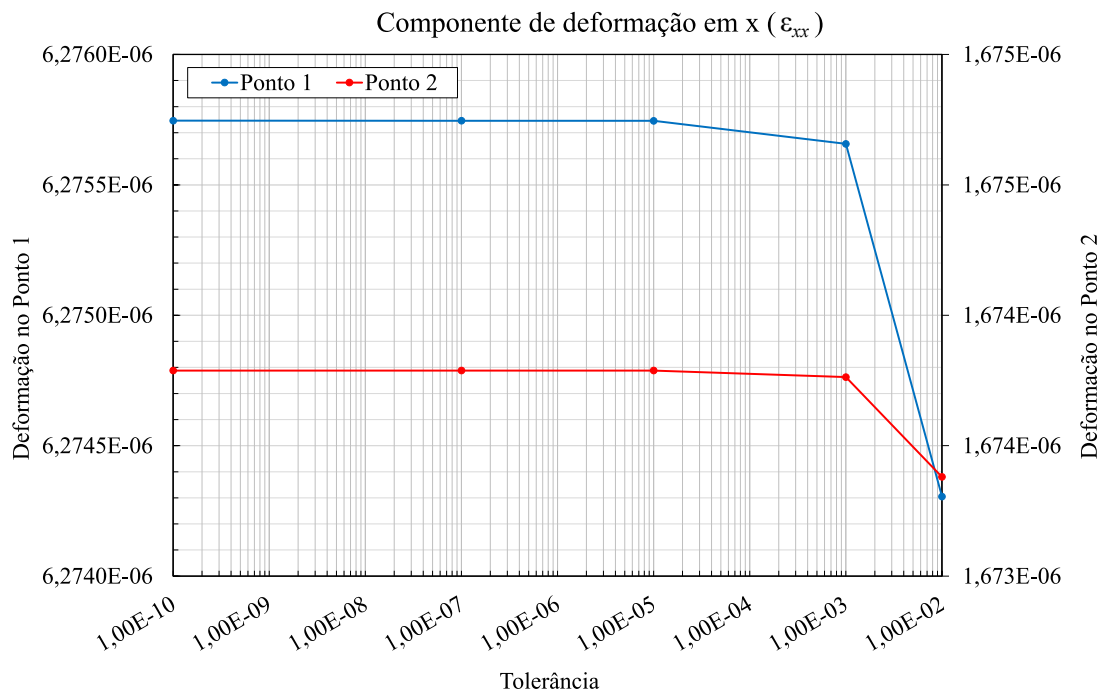


Figura 4.13: Componente de deformação ϵ_{xx} nos pontos 1 e 2 em função da tolerância adotada.

Tabela 4.3: Custo computacional do MEFG^{gl}-GLI para diferentes valores de tolerância (ϵ).

Resultados	Tolerância (ϵ)				
	10^{-2}	10^{-3}	10^{-5}	10^{-7}	10^{-10}
Número de iterações	3	4	6	7	10
Tempo de execução relativo	1,000	1,270	1,921	2,159	3,937

A Tabela 4.3 apresenta os valores medidos para as análises realizadas. Em geral, para o problema estudado, o aumento no número de iterações apresenta relação inversamente logarítmica com o valor de tolerância adotado. Isto é, a convergência demanda uma iteração a mais para cada redução de dez vezes no valor da tolerância.

Quanto ao tempo de execução, também se verifica forte relação entre a redução da tolerância e o aumento do tempo de execução. A Fig. 4.14 apresenta a relação obtida para o problema avaliado. É importante ressaltar que a tolerância aqui avaliada se refere ao GLI, procedimento utilizado para o acoplamento entre a escala global e a mesoescala. O procedimento de análise envolve outras etapas. Por essa razão, a relação entre a tolerância e o tempo de execução pode não ser direta. Por exemplo, o número de ciclos global-local na análise da mesoescala pelo MEFG^{gl} também impacta no tempo total de análise.

Há, portanto, significativo aumento no custo computacional conforme valores mais restritivos de tolerância são adotados. O uso de tolerância igual a 10^{-5} demanda aproximadamente duas vezes mais tempo do que a solução com tolerância de 10^{-2} .

Os resultados indicaram que não há ganho de qualidade significativo quando valores

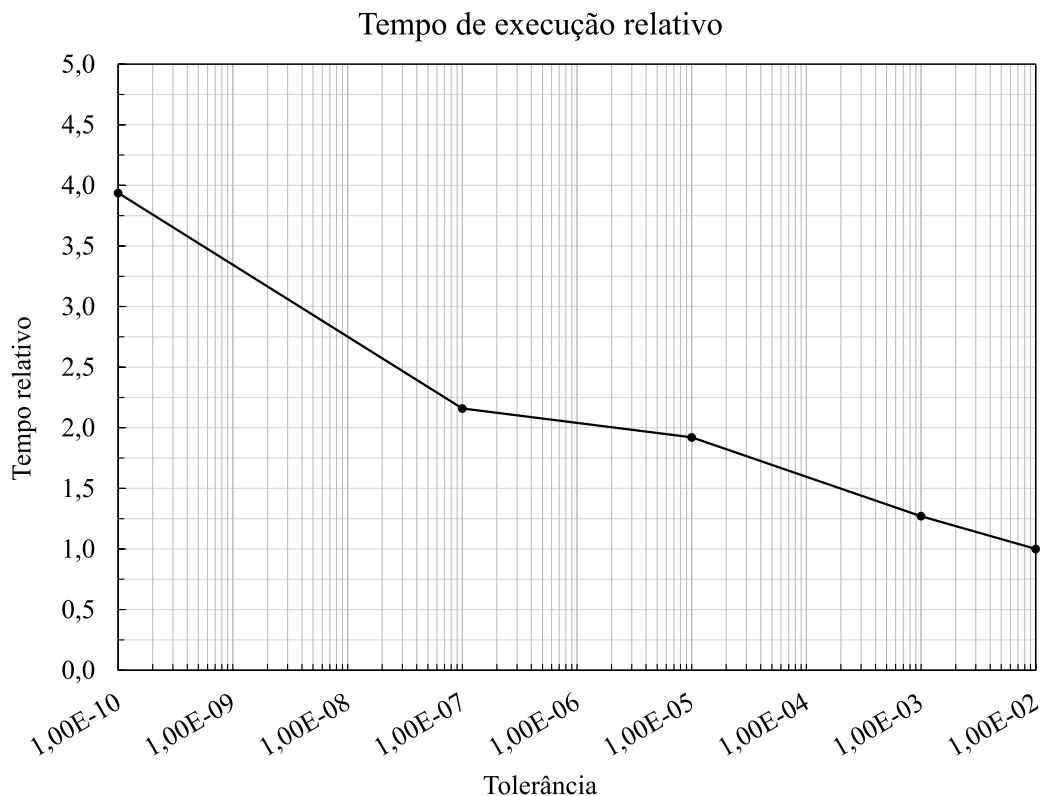


Figura 4.14: Tempo de execução para diferentes valores de tolerância.

de tolerância inferiores a 10^{-5} foram utilizados. Também verificou-se forte relação entre o uso de tolerâncias mais restritivas e o aumento do custo computacional. Nesse sentido, para a classe de problemas estudada, $\epsilon = 10^{-5}$ se apresenta como um valor equilibrado para a tolerância do resíduo relativo e, assim, foi adotado para a obtenção das demais soluções apresentadas neste texto.

4.3.2 Influência do número de ciclos MEFG^{gl}

No Algoritmo 3.1, o número de iterações global-local, ou ciclos MEFG^{gl} , é representado por n_{GL} e indica o número de vezes que as instruções apresentadas entre as Linhas 4 e 9 serão repetidas. Li, O'Hara e Duarte (2021) indicam que, no contexto do MEFG^{gl} -GLI, o uso de apenas um ciclo global-local é suficiente. Gupta, Kim e Duarte (2012) indicaram que uma iteração MEFG^{gl} é suficiente para lidar com problemas de condição de contorno inexatas considerando a classe de problemas em estudo.

Ainda assim, há interesse em investigar se o número de iterações MEFG^{gl} causa alguma influência no MEFG^{gl} -GLI. A estratégia de enriquecimento global-local aplicada ao MEFG^{gl} resulta em alterações das funções de forma do modelo global, o que é evidenciado na Eq. (2.11). Também ocorre imposição de condições de contorno no problema da escala local. Dentro do escopo do MEFG^{gl} -GLI, a mesoescala exerce o papel de escala global do MEFG^{gl} e, após o processo de enriquecimento, terá funções de forma construídas a partir de funções de enriquecimento global-local – novamente, conforme a Eq. (2.11). A matriz

de rigidez da mesoescala é, portanto, alterada devido ao processo de enriquecimento. A mesoescala, contudo, também é a escala local em um acoplamento pelo GLI, ainda dentro do escopo do $MEFG^{gl}$ -GLI. Recorda-se, então, que a diferença de rigidez entre os modelos da escala global e da mesoescala condiciona a convergência do GLI, como discutido na Seção 3.2.2 – ver Eq. (3.20). Nesse sentido, o número de ciclos $MEFG^{gl}$ pode, também, interferir no comportamento da convergência do GLI.

A influência do número de iterações global-local foi avaliada de forma similar à tolerância. O Problema 2 (ver Fig. 4.5) foi solucionado considerando 1, 2, 3 ou 6 iterações $MEFG^{gl}$.

A qualidade dos resultados foi avaliada pelo campo de deformações, tal como feito anteriormente. Novamente, os Pontos 1 e 2, apresentados na Fig. 4.12, foram considerados para a comparação dos resultados. A Fig. 4.15 apresenta a componente de deformação ϵ_{xx} resultante da solução pelo $MEFG^{gl}$ -GLI. Na figura, são apresentados os resultados obtidos com o menor e o maior número de iterações $MEFG^{gl}$. Em geral, os resultados indicam que as soluções obtidas são bastante similares. Para comparação, a Fig. 4.16 apresenta os resultados provenientes da solução pelo MEF convencional utilizando os Modelos A e B descritos anteriormente.

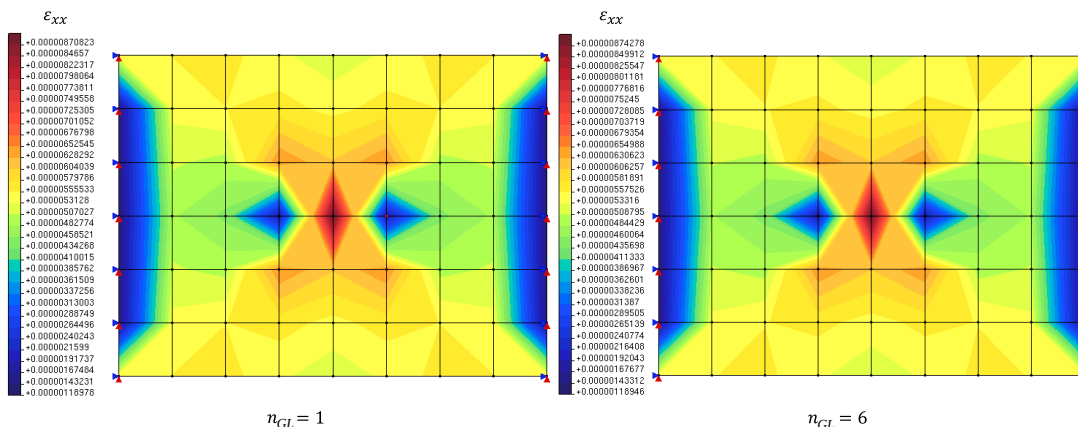


Figura 4.15: Componente de deformação ϵ_{xx} na mesoescala. Resultados obtidos pelo $MEFG^{gl}$ -GLI.

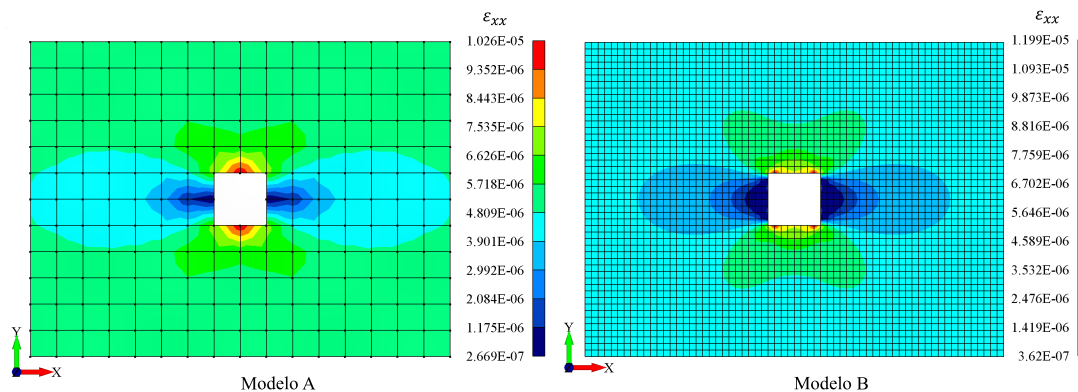


Figura 4.16: Componente de deformação ϵ_{xx} na mesoescala. Resultados obtidos pelo MEF.

A Tabela 4.4 apresenta de forma mais detalhada os resultados obtidos nos Pontos 1 e 2. Observa-se que há leve evolução de resultados ao se utilizar $n_{GL} = 2$ em comparação com o uso de apenas um ciclo MEF G^{gl} . A partir daí, não há melhora efetiva dos resultados, que se mantêm praticamente inalterados mesmo para $n_{GL} = 6$.

A influência do número de ciclos MEF G^{gl} na demanda computacional foi novamente avaliada pelo tempo de execução e pelo número de iterações necessárias para a convergência do GLI. A Tabela 4.5 apresenta os resultados obtidos. Independentemente do valor adotado para n_{GL} , o número de iterações GLI necessárias para a convergência se manteve inalterado e igual a 6.

O tempo de execução para $n_{GL} = 1$ foi adotado como referência. Observa-se que o aumento no número de ciclos MEF G^{gl} influenciou diretamente o tempo total da análise. No caso extremo, utilizando-se $n_{GL} = 6$, o tempo de execução foi 2,653 vezes superior ao obtido quando utilizado apenas uma iteração MEF G^{gl} . O gráfico da Fig. 4.17 apresenta relação entre n_{GL} e o tempo de execução do MEF G^{gl} -GLI obtida neste trabalho.

Por fim, ainda que o número de iterações GLI não tenha sofrido alterações em função de n_{GL} , foi avaliado se houve variação no comportamento da convergência do GLI em termos do resíduo relativo a cada iteração. Os resultados obtidos estão apresentados na Tabela 4.6 e indicam que, para a classe de problemas avaliada, o aumento do número de iterações global-local no MEF G^{gl} não influenciou a convergência do GLI.

Tabela 4.4: Resultados do MEF G^{gl} -GLI para diferentes números de ciclos MEF G^{gl} .

Ciclos (n_{GL})	Componentes de deformação (10^{-6})					
	Ponto 1			Ponto 2		
	ε_{xx}	ε_{yy}	ε_{xy}	ε_{xx}	ε_{yy}	ε_{xy}
1	6,2757	-2,6260	-0,8798	1,6738	-0,6710	0,0000
2	6,2801	-2,6003	-0,8802	1,6685	-0,6633	0,0000
3	6,2801	-2,6002	-0,8800	1,6685	-0,6633	0,0000
6	6,2801	-2,6002	-0,8800	1,6685	-0,6633	0,0000
Modelo A	6,1472	-2,4597	-1,0142	1,2127	-0,2659	0,0000
Modelo B	6,0636	-2,4028	-0,8580	1,6008	-0,4226	0,0000

Tabela 4.5: Custo computacional do MEF G^{gl} -GLI para diferentes números de ciclos MEF G^{gl} .

Resultados	Número de ciclos MEF G^{gl} (n_{GL})			
	1	2	3	6
Número de iterações MEF G^{gl} -GLI	6	6	6	6
Tempo de execução relativo	1,00	1,322	1,595	2,653

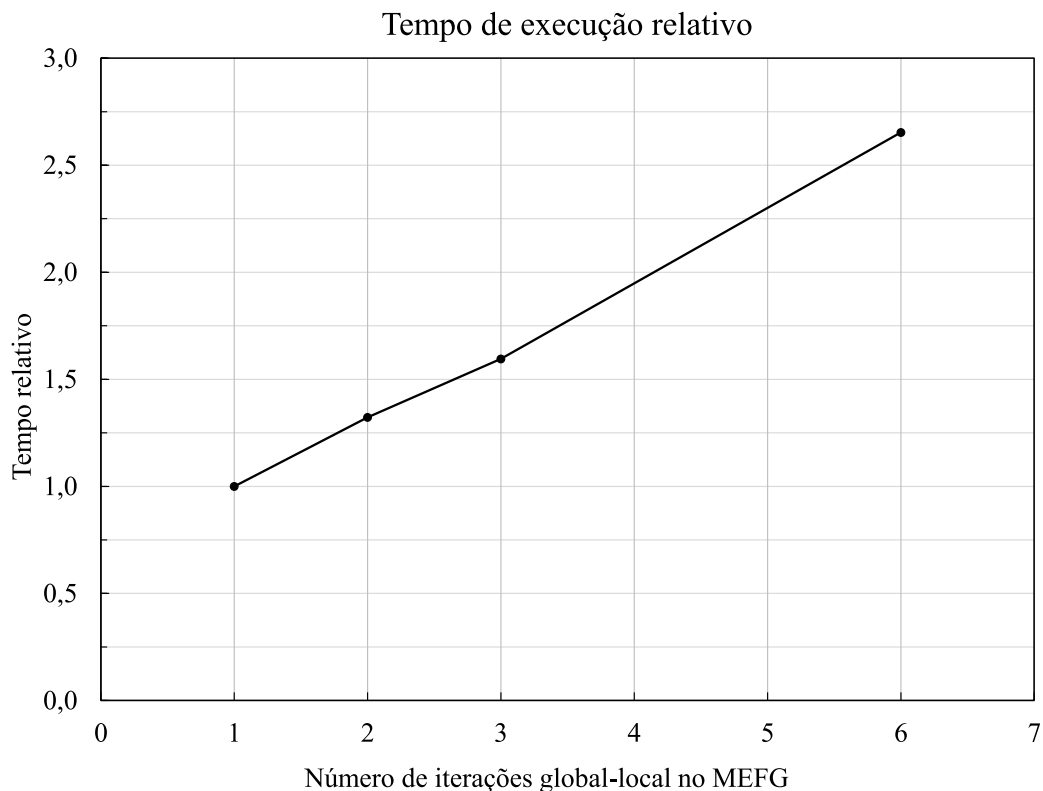


Figura 4.17: Tempo de execução para diferentes n_{GL} .

Em resumo, para o problema avaliado, o incremento no número de ciclos MEFG^{gl} não melhorou os resultados obtidos de modo significativo. Por outro lado, também não foram observadas alterações no comportamento de convergência do GLI com o aumento de n_{GL} . Constatou-se, entretanto, conforme esperado, que o tempo de execução da análise pode aumentar de forma significativa quando se utiliza elevado número de iterações na estratégia global-local aplicada ao MFG. Desse modo, neste trabalho, as próximas análises foram realizadas considerando $n_{GL} = 1$.

Tabela 4.6: Convergência do MEFG^{gl} -GLI para diferentes números ciclos MEFG^{gl} .

Iteração	Resíduo relativo			
	$n_{GL} = 1$	$n_{GL} = 2$	$n_{GL} = 3$	$n_{GL} = 6$
0	1,0000E+00	1,0000E+00	1,0000E+00	1,0000E+00
1	6,2556E-02	6,2526E-02	6,2524E-02	6,2524E-02
2	3,8612E-03	3,8620E-03	3,8618E-03	3,8618E-03
3	2,3807E-04	2,3829E-04	2,3827E-04	2,3827E-04
4	1,4677E-05	1,4701E-05	1,4700E-05	1,4700E-05
5	9,0486E-07	9,0697E-07	9,0686E-07	9,0688E-07

4.3.3 Influência da extensão da mesoescala

Duval, Passieux, Salaün e Guinard (2014) avaliaram a influência da extensão do domínio local na solução de problemas de mecânica da fratura pelo MEF-GLI. Não foi encontrado, entretanto, trabalho sobre a influência do tamanho da mesoescala em análises com o MEFG^{gl} -GLI. Justifica-se, assim, a investigação apresentada nesta seção.

Foram propostos, então, três modelos distintos para a solução do Problema 2. Tais modelos estão apresentados na Fig. 4.18 e diferenciam-se exclusivamente pelo domínio da mesoescala. No Modelo M1, a mesoescala possui duas camadas de elementos além do contorno do domínio da escala local. No Modelo M2, há apenas uma camada. Por fim, o Modelo M3 foi construído com a mesoescala coincidente com o domínio $\bar{\Omega}_L$. As demais características dos modelos foram mantidas iguais.

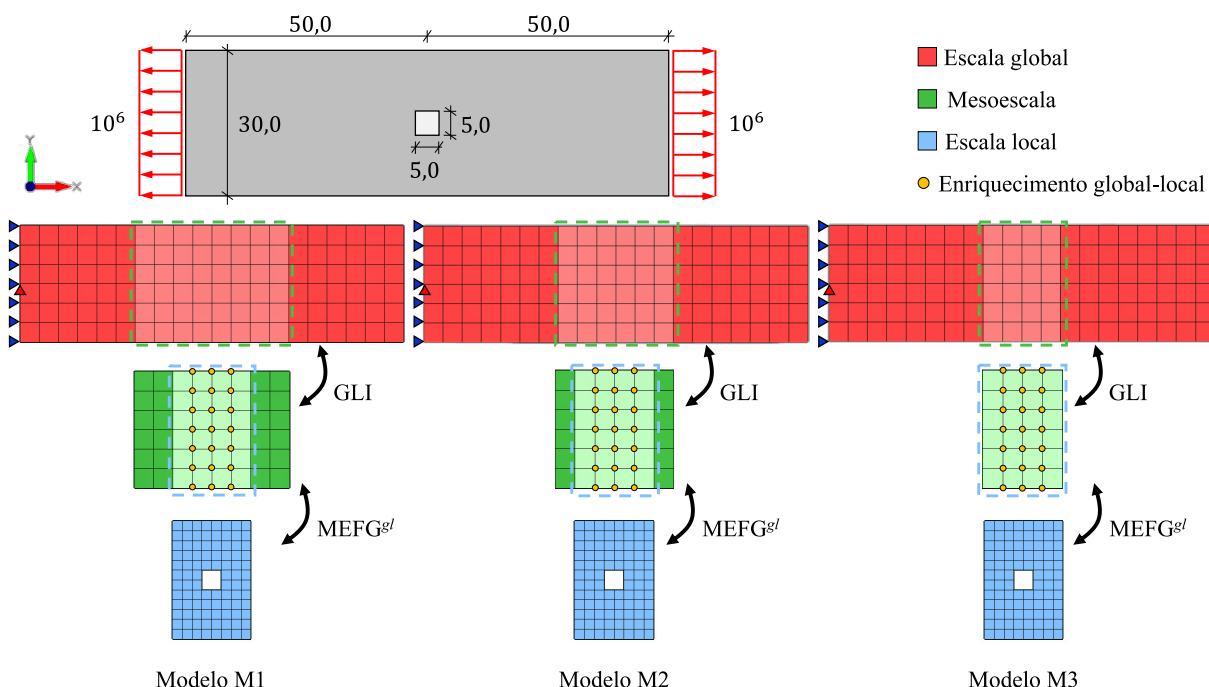


Figura 4.18: Modelos utilizados na avaliação da influência da abrangência da mesoescala nos resultados do MEFG^{gl} -GLI.

A qualidade dos resultados obtidos foi inicialmente avaliada em termos dos deslocamentos na direção y calculados para a borda inferior da chapa. A Fig. 4.19 apresenta um gráfico com as soluções obtidas pelos modelos M1, M2 e M3. Para referência, o gráfico inclui as soluções anteriormente obtidas pelos Modelos A e B (análise pelo MEF). Observa-se que ocorre quase uma sobreposição entre as respostas obtidas pelos Modelos M1, M2 e M3, que são muito similares à obtida pelo Modelo A.

Destaca-se que os resultados foram satisfatórios mesmo no Modelo M3, cujo domínio da mesoescala coincide com o domínio da escala local. Também se observa que os resultados não sofreram alterações independentemente da posição de Γ , a interface entre a escala global e a mesoescala. Esses resultados podem ser interpretados de forma positiva, pois

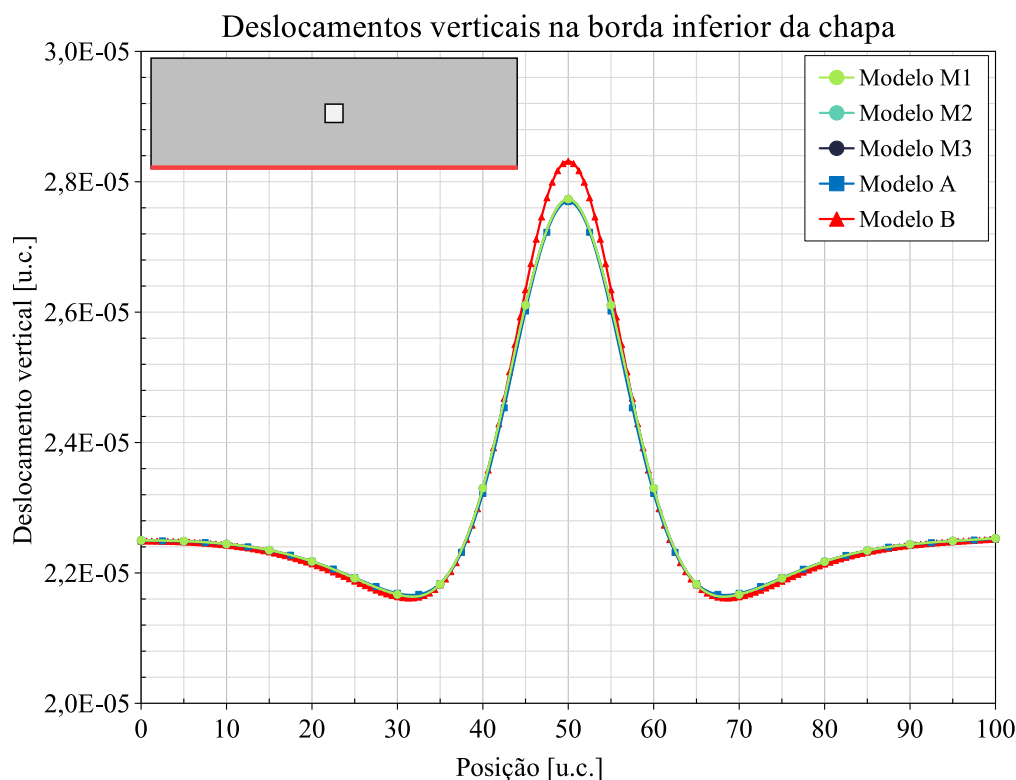


Figura 4.19: Deslocamentos verticais na borda inferior da chapa — Resultados dos Modelos M1, M2 e M3.

indicam que, para a classe de problemas avaliada, a definição do domínio da mesoescala não conduziu a resultados espúrios em Γ em qualquer dos cenários avaliados.

Também foi avaliado se a extensão da mesoescala influencia na demanda computacional para a obtenção da solução. Por um lado, os Modelos M2 e M3 possuem a mesoescala reduzida em relação ao modelo M1, o que, em princípio, reduz o tempo de execução. Por outro lado, é necessário avaliar se a convergência do GLI é prejudicada ao se aproximar Γ do domínio local.

Os resultados obtidos estão apresentados na Tabela 4.7, na qual o tempo de execução associado ao Modelo M1 foi tomado como referência. O Modelo M2 apresentou leve redução no tempo de solução, o que pode ser justificado pela redução do tamanho do modelo da mesoescala. Para o Modelo M3, entretanto, ocorreu o contrário. O tempo de solução foi maior do que o gasto nos Modelos M1 e M2. Isso ocorreu devido ao número de iterações GLI necessárias para o atendimento do critério de convergência, que aumentou de 6, nos Modelos M1 e M2, para 8 no Modelo M3.

A abrangência da mesoescala pode, portanto, influenciar a convergência do $MEFG^{gl}$ -GLI. Essa afirmação é evidenciada na Fig. 4.20, que apresenta o comportamento da convergência dos Modelos M1, M2 e M3. Os resultados indicam de forma clara que a taxa de convergência do GLI é prejudicada quando Γ se aproxima da região sob influência dos efeitos localizados.

Tabela 4.7: Custo computacional do MEFG^{gl}-GLI para diferentes tamanhos da mesoescala.

Resultados	Modelo		
	M1	M2	M3
Número de iterações MEFG ^{gl} -GLI	6	6	8
Tempo de execução relativo	1,000	0,981	1,250

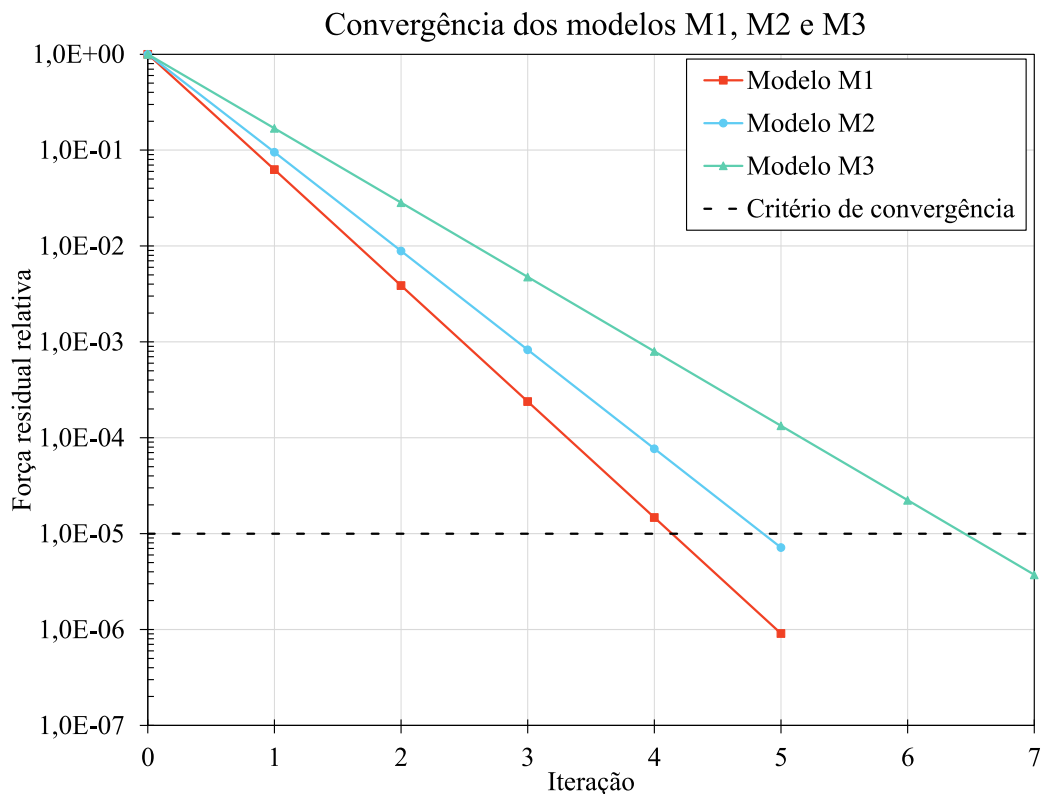


Figura 4.20: Convergência dos Modelos M1, M2 e M3.

Em resumo, a avaliação do Problema 2 por meio dos Modelos M1, M2 e M3 indicou que todos os modelos apresentaram a mesma qualidade de solução. Em relação ao tempo de solução, há uma troca entre a redução do problema matemático associado à mesoescala e o aumento no número de iterações GLI. Esse último comportamento ocorre, pois, a redução da mesoescala influenciou a taxa de convergência do MEFG^{gl}-GLI.

4.3.4 Técnicas de aceleração

Avaliou-se, por fim, o desempenho de algumas técnicas de aceleração apresentadas na Seção 3.3. Foram avaliadas as técnicas de relaxação estática e relaxação dinâmica. Para tal, o Problema 2 foi modificado. Como discutido anteriormente, a taxa de convergência do GLI é estabelecida a partir da diferença de rigidez entre a escala global e a mesoescala enriquecida. Quando $\bar{\Omega}_L$ é representado de forma menos rígida no modelo global, em

teoria, o GLI pode não convergir. Buscou-se, então, introduzir esse cenário no Problema 2.

Para tal, os modelos da mesoescala e da escala local foram alterados de modo que o módulo de elasticidade de todos os seus elementos fosse três vezes maior do que o utilizado na escala global. O modelo global possui módulo de elasticidade $E_G = 2,0 \times 10^{11}$ u.c., enquanto os modelos da mesoescala e da escala local possuem módulo de elasticidade $E_M = 6,0 \times 10^{11}$ u.c.. Esse problema será referido como Problema 3.

As soluções apresentadas a seguir foram obtidas mantendo-se os mesmos parâmetros do MEF G^{gl} -GLI, definidos a partir das observações apresentadas nas seções anteriores deste capítulo. Isto é, foi utilizado $\epsilon = 10^{-5}$ e $n_{GL} = 1$. Em relação à extensão da mesoescala, foi utilizada a correspondente ao Modelo M1, portanto, com duas camadas de elementos entre a interface Γ e $\partial\Omega_L$.

Relaxação Estática

Inicialmente, tentou-se resolver o problema sem a utilização de qualquer técnica de aceleração. Conforme esperado a partir das discussões da Seção 3.2.2, essa tentativa não logrou sucesso, tendo sido observado desde o início do procedimento que a convergência que não seria atingida.

Diversos fatores de relaxação ω foram, então, utilizados em busca da solução do Problema 3. A evolução da convergência obtida para cada um desses fatores está apresentada na Fig. 4.21. Também foram incluídos no gráfico os resultados das primeiras iterações obtidas sem a utilização de relaxação, o que é equivalente ao uso de $\omega = 1, 0$.

É evidente que o emprego da técnica de relaxação estática possibilitou a solução do Problema 3 por meio do MEF G^{gl} -GLI. Os resultados indicam que a taxa de convergência depende fortemente do valor de relaxação utilizado.

Quanto ao custo computacional, observou-se relação entre o número de iterações necessárias para a convergência e o coeficiente de relaxação utilizado. A variação do número de iterações em função de ω está apresentada na Fig. 4.22.

No problema avaliado, no qual não há distorção dos elementos finitos, observou-se relação inversamente proporcional entre a razão E_M/E_G e os valores de coeficientes de relaxação que conduziram às melhores taxas de convergência. O módulo de elasticidade utilizado na mesoescala foi 3 vezes superior ao da escala global. O orifício, contudo, representa uma perda de rigidez, que é considerada apenas na mesoescala por meio do MEF G^{gl} . A razão entre a rigidez do modelo da mesoescala e da escala global é, portanto, pouco menor que 3,0. As melhores taxas de convergência foram obtidas por valores de ω próximos de 0,35.

Quanto à qualidade dos resultados obtidos pela relaxação estática, as soluções foram avaliadas em termos dos deslocamentos e das deformações. Para comparação, o Modelo B, apresentado anteriormente, foi atualizado para o Problema 3. O módulo de elasticidade

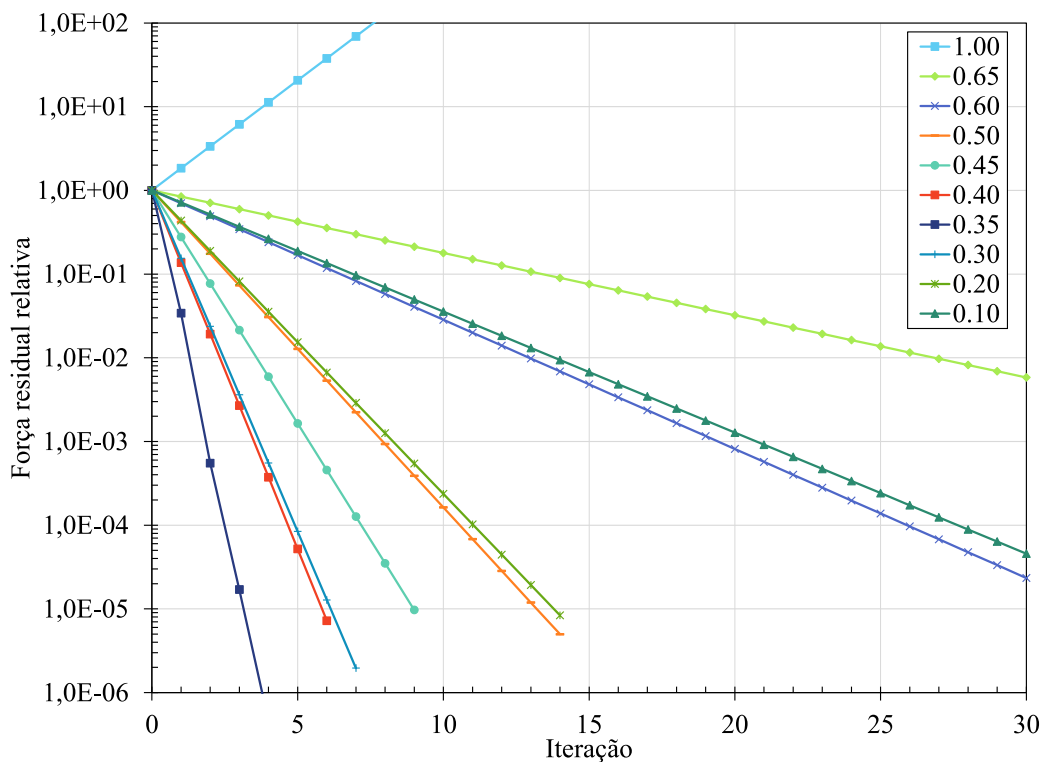


Figura 4.21: Convergência utilizando relaxação estática com diferentes valores de ω .

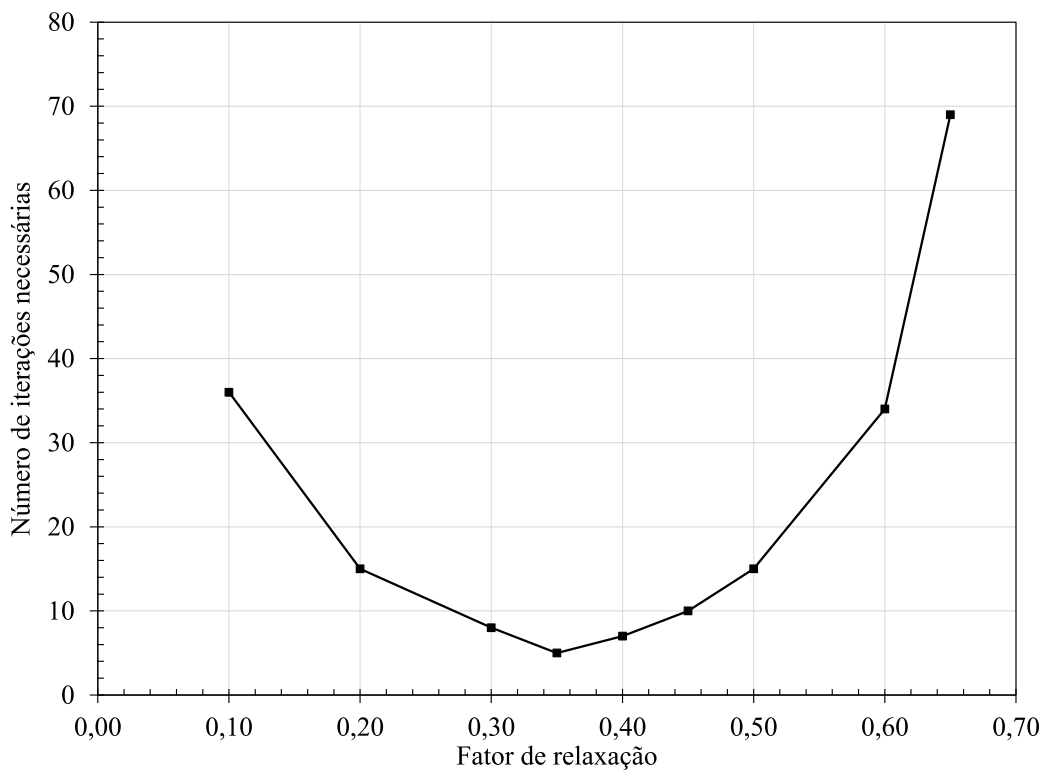


Figura 4.22: Relação entre o número de iterações MEFG^{gl}-GLI e ω .

foi, portanto, modificado somente no domínio correspondente à mesoescala da análise MEF gl -GLI e foi utilizado o valor $E_M = 6,0 \times 10^{11}$ u.c., isto é, valor três vezes maior do que o valor original.

Em geral, as soluções obtidas utilizando a relaxação estática não apresentam diferenças entre si, independentemente do fator de relaxação utilizado. Isso será explorado com maiores detalhes adiante no texto. Dito isso, escolheu-se a solução obtida com $\omega = 0,35$ como representativa. A Fig. 4.23 apresenta os deslocamentos u_x e u_y e as componentes de deformação ε_{xx} e ε_{yy} obtidos pelo MEF gl -GLI com relaxação estática utilizando $\omega = 0,35$. Para referência, os resultados obtidos pelo Modelo B estão apresentados na Fig. 4.24.

Percebe-se que os campos de deslocamento e de deformação obtidos utilizando a relaxação estática se assemelham aos obtidos pelo MEF convencional. Os Pontos 1 e 2 (ver Fig. 4.12) foram novamente utilizados para a comparação dos resultados obtidos em termos das componentes de deformações. A Tabela 4.8 apresenta os resultados obtidos com alguns dos fatores de relaxação utilizados. Selecionou-se o maior fator de relaxação ($\omega = 0,65$), o menor ($\omega = 0,10$) e o fator com melhor taxa de convergência ($\omega = 0,35$).

Os resultados apresentados na Tabela 4.8 indicam que, em geral, o uso da relaxação estática não introduziu resultados espúrios. É necessário comentar, contudo, que a solução que utilizou $\omega = 0,65$ encontrou um valor não nulo para ε_{xy} no Ponto 2, o que não ocorre nas demais soluções. Esse erro é marginal. Ademais, as soluções obtidas pelo MEF gl -GLI com relaxação estática são próximas da obtida pelo Modelo B.

Quanto aos deslocamentos, a Fig. 4.25 apresenta os valores de u_y obtidos na borda inferior da chapa pelo MEF, por meio do modelo B, e pelo MEF gl -GLI utilizando $\omega = 0,35$. A resposta obtida nos dois modelos é bastante similar e, portanto, os resultados do MEF gl -GLI são satisfatórios. Recobra-se que o Modelo B possui 15.680 graus de liberdade, enquanto a solução pelo MEF gl -GLI foi obtida com 690 graus de liberdade. Na borda inferior, a maior diferença entre os deslocamentos obtidos é de 2,69% e ocorre em $x = 50,0$ u.c..

Tabela 4.8: Resultados do MEF gl -GLI para alguns valores de ω .

Fator de relaxação (ω)	Componentes de deformação (10^{-6})					
	Ponto 1			Ponto 2		
	ε_{xx}	ε_{yy}	ε_{xy}	ε_{xx}	ε_{yy}	ε_{xy}
0,65	2,0219	-0,8200	-0,2757	0,5157	-0,1682	0,0001
0,35	2,0219	-0,8199	-0,2757	0,5157	-0,1682	0,0000
0,10	2,0219	-0,8199	-0,2757	0,5157	-0,1682	0,0000
Modelo B	1,9712	-0,7587	-0,2637	0,5105	-0,1059	0,0000

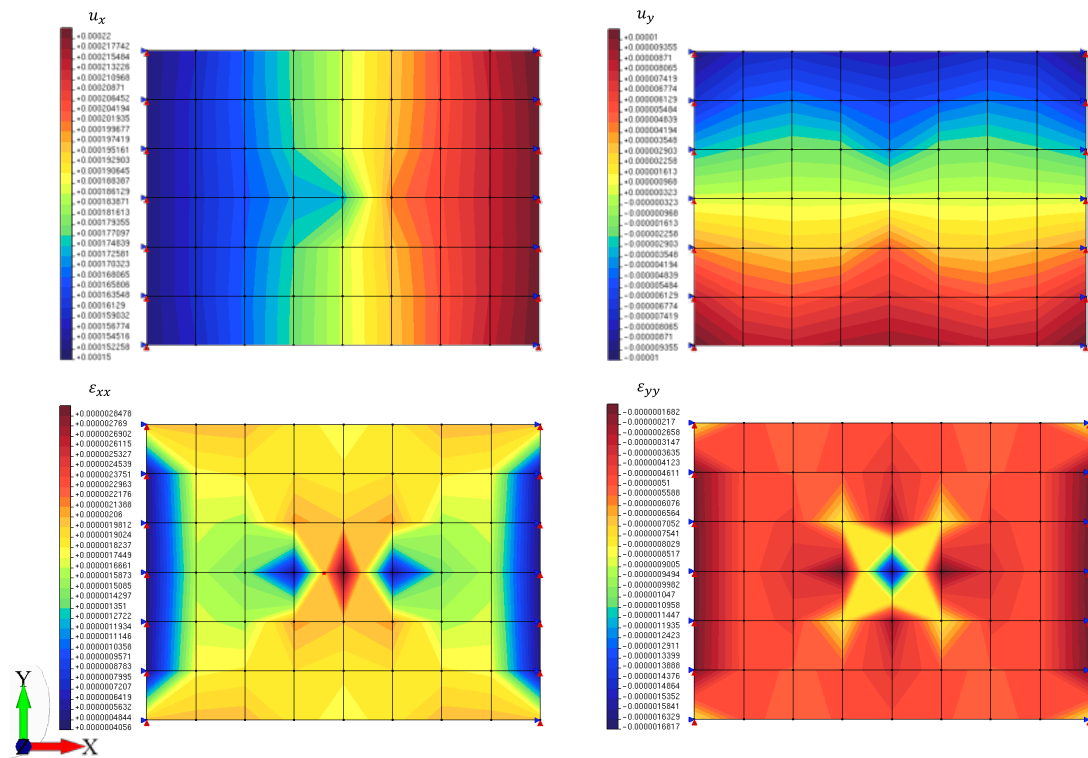


Figura 4.23: Deslocamentos e deformações obtidas pelo MEF gl -GLI utilizando relaxação estática com $\omega = 0,35$.

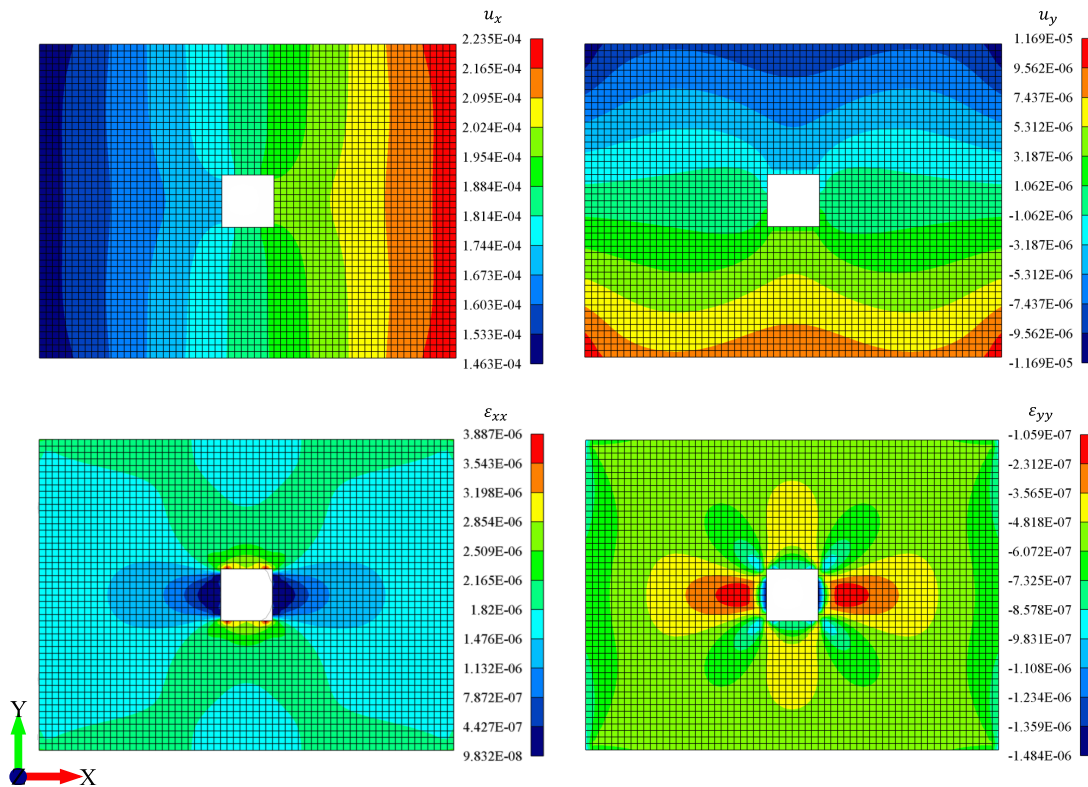


Figura 4.24: Deslocamentos e deformações obtidas pelo Modelo B.

A seção transversal no plano yz localizada em $x = 45,0$ u.c. foi selecionada para a avaliação dos deslocamentos horizontais. Os resultados são apresentados na Fig. 4.26 e, novamente, indicam que o MEF gl -GLI com relaxação estática conduz a resultados ajustados aos do Modelo B. A maior diferença entre as respostas ocorre em $y = 15,0$ u.c. e é de 0,57%.

A relaxação estática mostrou-se eficiente na tarefa de permitir a solução de problemas nos quais a convergência não ocorre sem o uso dessa técnica. A convergência é estável e segue taxa logarítmica. Houve leve prejuízo nos resultados quando foi utilizado fator de relaxação que levou à convergência muito lenta ($\omega = 0,65$). A determinação de bons valores de ω é justamente a maior limitação da técnica. Pode não ser fácil saber, *a priori*, qual valor de ω é adequado. Ainda que o presente estudo tenha indicado a existência de uma relação entre bons valores de ω e a razão entre as rigidezes da mesoescala e da escala global, é necessário validar essa observação a partir da avaliação de outras classes de problemas. Não obstante, nem sempre é fácil identificar a relação de rigidez entre as escalas.

Outra limitação da relaxação estática é a própria característica do fator ser constante ao longo de todo o processo. Desse modo, se houver alguma mudança no problema da mesoescala ou da escala local, como o surgimento de um contato, a propagação de trinca ou até mesmo efeitos de plasticidade, um bom valor de ω no início da análise pode passar a não ser eficiente.

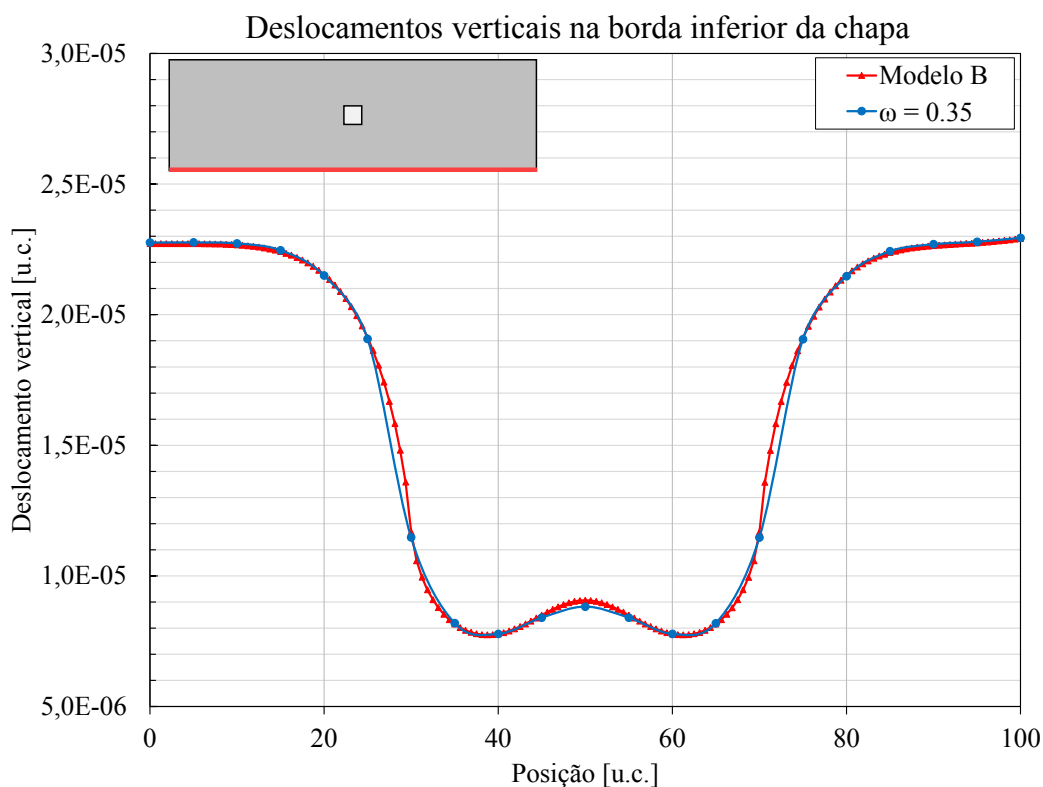


Figura 4.25: Deslocamentos verticais na borda inferior da chapa pelo Modelo B (MEF) e pelo MEF gl -GLI com $\omega = 0,35$.

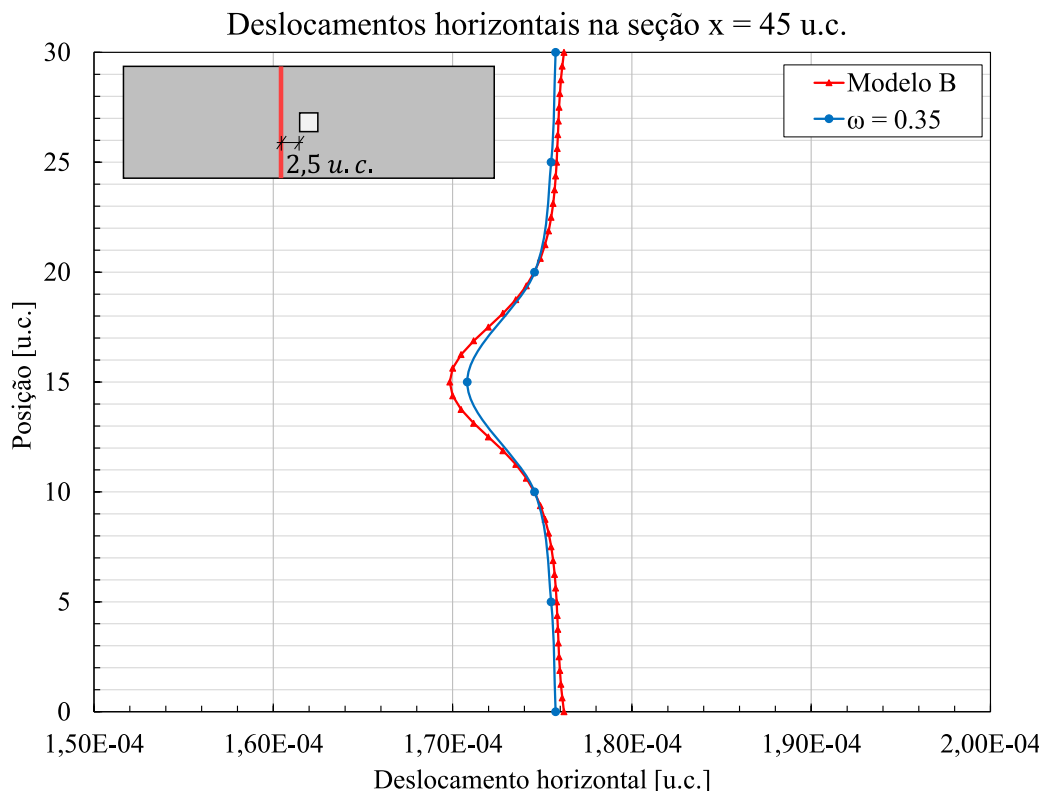


Figura 4.26: Deslocamentos u_x em $x = 45,0$ u.c. obtidos pelo Modelo B (MEF) e pelo MEFGLI com $\omega = 0,35$.

Relaxação Dinâmica

O Problema 3 foi novamente solucionado, agora aplicando-se a técnica de relaxação dinâmica, que foi utilizada de duas formas. Foi realizada uma solução conforme a sugestão presente na literatura, utilizando-se $\omega^0 = \omega^1 = 1,0$. Também foi realizada uma segunda solução considerando $\omega^0 = \omega^1 = 0,35$. O objetivo é avaliar se há alguma vantagem em permitir que a relaxação dinâmica seja aplicada com valor inicial diferente de 1,00, aproveitando, assim, algum conhecimento *a priori* sobre o problema. Os resultados obtidos são apresentados na Fig. 4.27.

Em relação à relaxação dinâmica com $\omega^0 = \omega^1 = 1,00$, observa-se que nas duas primeiras iterações o valor do resíduo relativo é igual ao obtido quando se utiliza relaxação estática com $\omega = 1,00$. Isso está de acordo com o Algoritmo 3.3. A partir da terceira iteração, a convergência segue com taxa muito similar à obtida quando a relaxação estática foi aplicada com $\omega = 0,35$.

Quando a relaxação dinâmica foi utilizada com $\omega^0 = \omega^1 = 0,35$, foi obtido resultado muito próximo ao da relaxação estática com $\omega = 0,35$. Cabe salientar que o Problema 3 não possui não linearidades, de modo que o comportamento de todas as escalas é linear.

A qualidade dos resultados obtidos foi avaliada em termos dos deslocamentos u_y na borda inferior da chapa. A Fig. 4.28 apresenta os resultados obtidos pelo MEFGLI utilizando a relaxação dinâmica e a relaxação estática com $\omega = 0,35$, bem como está

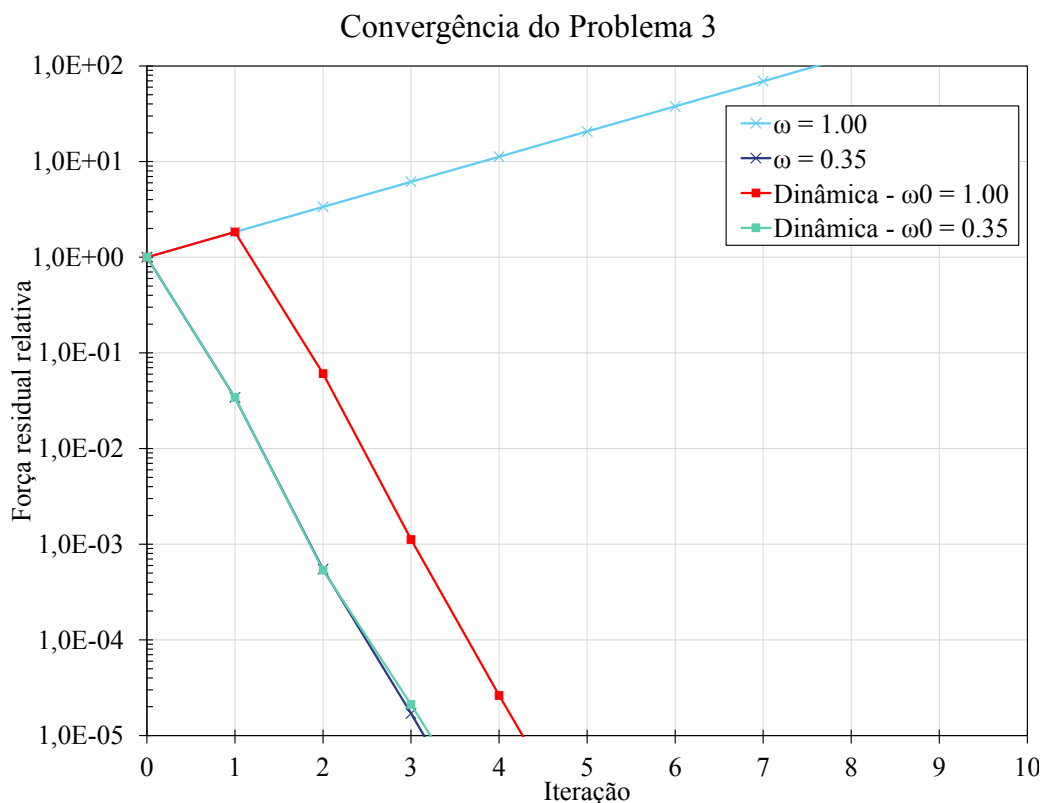


Figura 4.27: Convergência do MEFG^{gl}-GLI utilizando diferentes técnicas de aceleração.

apresentada a resposta obtida pelo Modelo B. Não foi observado prejuízo nas soluções obtidas com a relaxação dinâmica em comparação às demais. De fato, para a tolerância considerada, os resultados são coincidentes até que várias casas decimais sejam consideradas.

Em seguida, a relaxação dinâmica foi aplicada na solução de novos problemas derivados do Problema 2. Se o Problema 3 foi definido como a modificação do Problema 2 na qual o módulo de elasticidade da mesoescala e da escala local era 3 vezes superior o módulo de elasticidade da escala global, foram definidos novos problemas nos quais essa relação fosse 4, 6 e 10 vezes. Esses quatro problemas foram resolvidos utilizando relaxação dinâmica. Dessa vez, somente foi considerado $\omega^0 = \omega^1 = 1,00$. Os resultados obtidos estão apresentados na Fig. 4.29. No gráfico, as curvas nE referem-se ao problema em que a relação E_M/E_G vale n .

Os resultados indicam que, nessa classe de problemas, isto é, linear elástico, o uso da relaxação dinâmica conduziu à taxas de convergência semelhantes independentemente da diferença de rigidez entre as escalas. Quando a relaxação dinâmica é utilizada, portanto, não se verifica o preestabelecimento da taxa de convergência em função da diferença de discretização de $\bar{\Omega}_L$ em cada uma das escalas. Essa característica contorna uma das principais limitações do GLI.

A Tabela 4.9 apresenta o fator de relaxação ω^k a cada iteração k para diferentes valores da razão n entre a rigidez das escalas. Para comparação, os problemas 4E, 6E e 10E

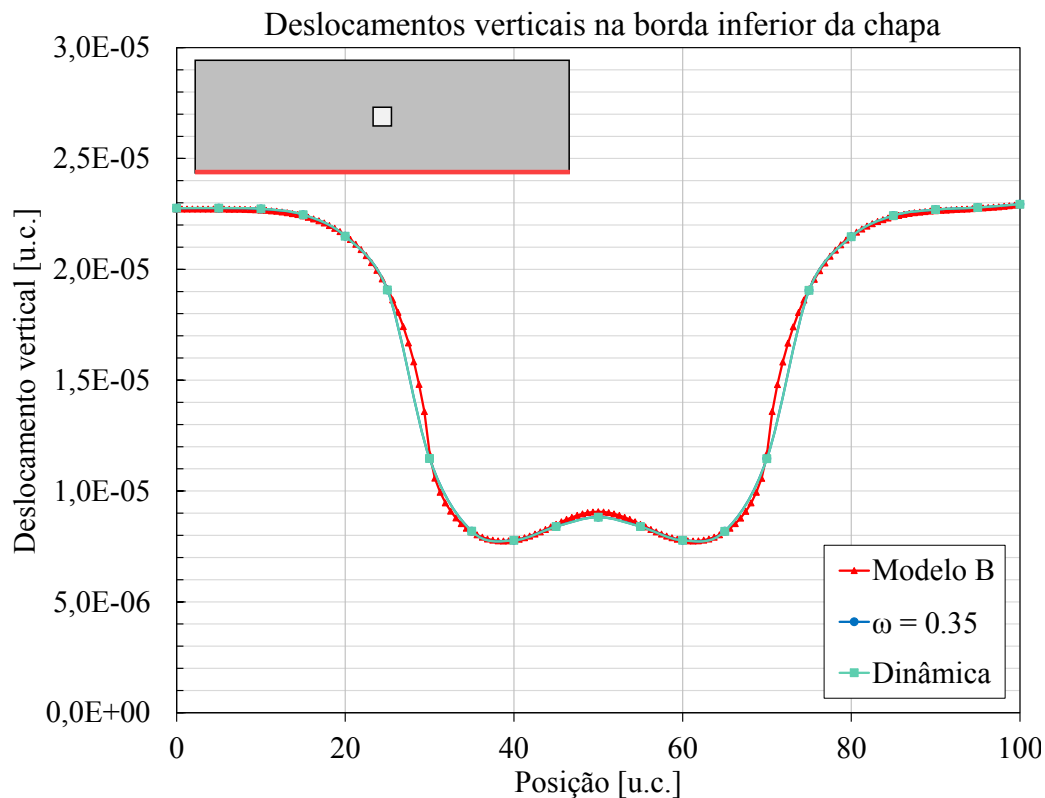


Figura 4.28: Deslocamentos verticais na borda inferior da chapa obtidos pelo Modelo B (MEF) e o e pelo MEFG^{gl}-GLI com $\omega = 0,35$ e com relaxação dinâmica.

também foram resolvidos utilizando relaxação estática com diferentes fatores de relaxação ω . Em cada caso, ω_{est} denota o fator que conduziu mais rapidamente à convergência. Observa-se que, a partir da terceira iteração, o valor de ω^k tende a $1/n$. Considerando a classe do problema e as características do método Δ^2 de Aitken, esses resultados reforçam as impressões sobre a existência de uma correlação entre bons valores de ω e n .

Tabela 4.9: Fator de relaxação por iteração para diferentes valores de $n = E_M/E_G$.

Iteração	Fator de relaxação (ω^k)			
	$n = 3$	$n = 4$	$n = 6$	$n = 10$
0	1,000	1,000	1,000	1,000
1	1,000	1,000	1,000	1,000
2	0,354	0,266	0,177	0,106
3	0,352	0,264	0,176	0,106
4	0,342	0,3257	0,171	0,103
5	0,346	0,259	0,173	0,104
ω_{est}	0,350	0,250	0,170	0,100

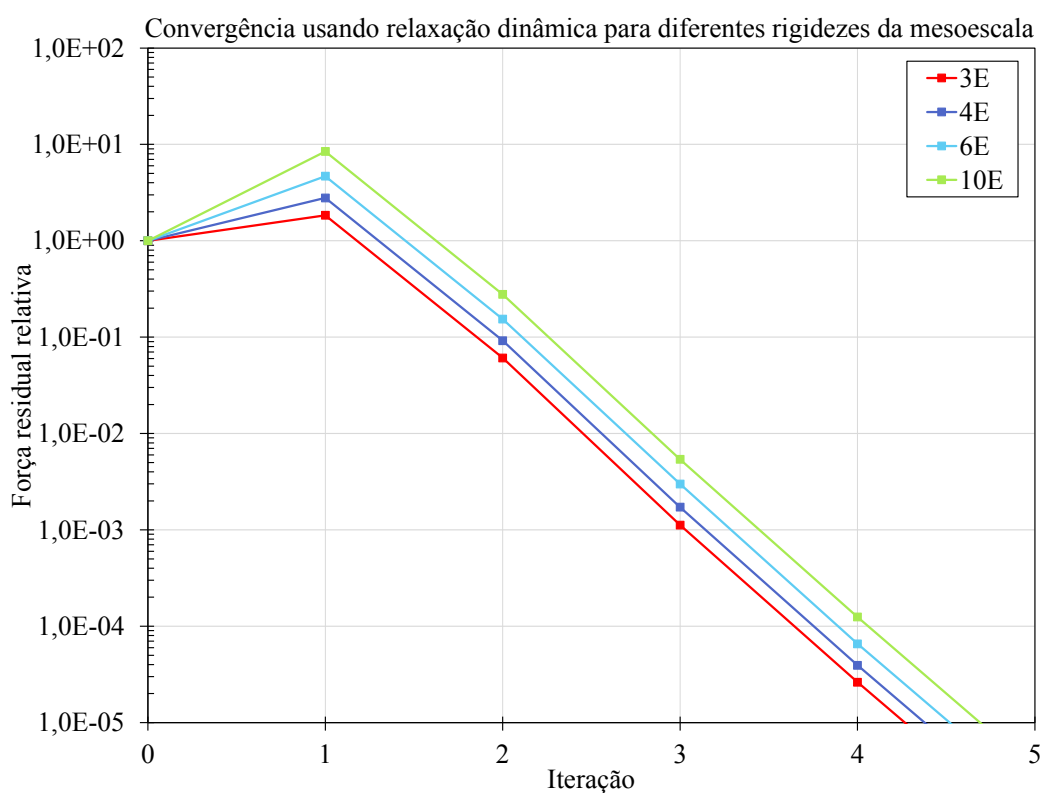


Figura 4.29: Convergência utilizando relaxação dinâmica para diferentes rigidezes da mesoescala.

Capítulo 5

Considerações finais

5.1 Contribuições

Neste trabalho, foi realizada a implementação de um acoplamento não intrusivo de duas escalas de análise. Em cada uma dessas escalas, o problema estrutural é simulado por meio de diferentes programas de análise. Ainda que possa ser utilizada para esse fim, isto é, acoplar resultados entre dois modelos de elementos finitos com malha coincidente na interface, a implementação foi direcionada para a estratégia proposta por Li, O’Hara e Duarte (2021) denominada $MEFG^{gl}$ -GLI.

A estratégia de acoplamento foi inicialmente aplicada para a solução manual de um problema simples. Após a validação desse resultado, foi elaborado um código para a aplicação do $MEFG^{gl}$ -GLI na plataforma **INSANE**. Os resultados foram validados por meio da comparação da solução de problemas estruturais de comportamento linear elástico com resultados obtidos pelo MEF convencional.

Um dos objetivos deste trabalho era apresentar uma via de aproximação entre programas de análise estrutural utilizados em meio acadêmico, como o **INSANE**, e pacotes computacionais amplamente empregados na indústria. Nesse sentido, uma vez que a implementação do $MEFG^{gl}$ -GLI no **INSANE** estava validada, o código foi expandido para permitir que o *software* Abaqus fosse utilizado como solucionador da escala global pelo MEF. A plataforma **INSANE** continuou sendo responsável pela solução da mesoescala e da escala local por meio da estratégia global-local aplicada ao MEF ($MEFG^{gl}$). Essa implementação passou pelos mesmos testes de validação utilizados anteriormente.

O código gerenciador do acoplamento foi concebido tendo em vista a sua expansão para integração com outros programas. Desse modo, o programa foi escrito em Python, uma linguagem de programação de alto nível, com biblioteca extensa e que é frequentemente utilizada nas APIs de programas de análise pelo MEF, como o Abaqus, Ansys, Femap e SAP. O código foi construído segundo o paradigma de programação orientada à objetos e foram utilizadas classes abstratas para facilitar eventuais expansões do código para torná-

lo compatível com outros programas de análise. Todo o material produzido está disponível no link: <https://github.com/neimarsilveira/gfemgl-gli.git>.

As investigações realizadas buscaram tanto confirmar conclusões disponíveis na literatura, quanto obter novas informações sobre o procedimento. Vale salientar que todas as investigações realizadas neste trabalho se referem à classe de problemas com comportamento linear elástico.

O primeiro parâmetro do MEFG^{gl} -GLI investigado foi a tolerância do processo iterativo de solução pelo GLI. Foi observado que o uso de valores de tolerância rigorosos causam significativo aumento no custo computacional da solução. O uso de tolerância $\epsilon = 10^{-10}$ consome aproximadamente quatro vezes mais tempo do que a solução com tolerância igual a 10^{-2} . Para a classe de problemas avaliada, não foi observado ganho na qualidade de aproximação para valores de tolerância superiores a 10^{-5} , de modo que se sugere que esse valor seja utilizado para a solução de tipo de problema. Essa observação está de acordo com o indicado por Li, O'Hara e Duarte (2021).

Em seguida, foi avaliada a influência do número de iterações MEFG^{gl} (n_{GL}) na resposta obtida pelo MEFG^{gl} -GLI. Verificou-se que o incremento no número de iterações MEFG^{gl} não resultou em melhoria significativa das aproximações obtidas. Também não se observou alterações no comportamento da convergência do MEFG^{gl} -GLI com o aumento de n_{GL} . Constatou-se, entretanto, como era esperado, que o tempo de execução da análise pode aumentar de forma significativa quando se utiliza muitos ciclos MEFG^{gl} . Desse modo, para problemas com característica similar ao apresentado neste trabalho, recomenda-se, assim como em Li, O'Hara e Duarte (2021), o uso de $n_{GL} = 1$.

A influência da extensão do domínio da mesoescala também foi estudada. A qualidade da aproximação obtida não sofreu alterações significativas com a redução da mesoescala. Constatou-se, contudo, influência na demanda computacional do procedimento. Isso já era esperado, afinal quanto maior o modelo da mesoescala, mais recursos serão necessários para solucioná-lo. Porém, os estudos realizados neste trabalho indicaram que há uma outra relação, com efeito contrário. A redução demasiada da mesoescala, aproximando a interface Γ e $\partial\Omega_L$, influencia as forças residuais calculadas em Γ e, assim, prejudica a taxa de convergência do MEFG^{gl} -GLI. Há, portanto, uma troca entre a redução do problema matemático associado à mesoescala, que diminui a demanda computacional, e o aumento do número de iterações GLI, que aumenta a demanda computacional.

Por fim, foram avaliadas algumas técnicas de aceleração da convergência para o MEFG^{gl} -GLI. Conforme explorado neste texto, a necessidade do uso de tais técnicas decorre da dependência da taxa de convergência do GLI em relação à diferença de rigidez dos modelos da mesoescala e da escala global. Quanto maior essa diferença, mais lenta será a convergência, que pode ainda não ser possível caso o domínio local seja suficientemente mais rígido na mesoescala do que na escala global.

A técnica de relaxação estática mostrou-se eficiente. Sua implementação, bastante

simples, não onera o procedimento e possibilita a solução de problemas cuja convergência não seria possível sem o uso da técnica. Não se verificou prejuízo nos resultados obtidos devido ao uso da relaxação estática em comparação com as soluções obtidas pelo MEF convencional. A taxa de convergência é bastante estável e segue o comportamento o logarítmico característico do GLI.

Também foi implementada e avaliada a técnica de relaxação dinâmica. Observou-se que, para o tipo de problema estudado, ao longo das iterações, o fator de relaxação se aproximava daquele que tinha obtido o melhor desempenho na relaxação estática. Essa observação motivou a aplicação da relaxação dinâmica para a solução de problemas com diferentes valores de $n = E_M/E_G$, a razão entre os módulos de elasticidade da mesoescala e da escala global. Os resultados indicaram que os valores do fator de relaxação ω^k tendiam a $1/n$ com o avanço das iterações k . Endossa-se que esse comportamento foi observado em problemas lineares elásticos e que é necessário realizar investigações mais profundas acerca do assunto. Ainda, observou-se que o uso da relaxação dinâmica para a solução de problemas lineares elásticos faz com que a diferença de rigidez entre as escalas acopladas pelo GLI deixe de governar a taxa de convergência da solução — o que é uma das limitações do GLI. Essa observação veio da aplicação da relaxação dinâmica para a solução de um mesmo problema para vários valores de $n = E_M/E_G$. Independentemente do valor de n , todos os problemas apresentaram a mesma taxa de convergência quando utilizada a técnica de relaxação dinâmica.

5.2 Sugestões para trabalhos futuros

A maior parte do esforço realizado neste trabalho esteve voltado à implementação da estratégia de solução denominada MEFG^{gl}-GLI com base na proposta de Li, O'Hara e Duarte (2021). Também se buscou a confirmação de algumas observações da literatura, bem como foram realizadas investigações complementares.

Há, contudo, um vasto campo de investigações e de contribuições acerca dessa estratégia. Na lista a seguir, são apresentadas sugestões de linhas de estudo relacionadas ao tema:

1. Aplicar a estratégia para a solução de problemas com maior proximidade da realidade da engenharia, nos quais a complexidade (ou a extensão) da escala global justifique, de fato, a utilização de um programa com apelo industrial.
2. Reavaliar os parâmetros do MEFG^{gl}-GLI (ver Seção 4.3) na solução de outras classes de problemas, por exemplo, problemas com não linearidade física ou geométrica, problemas com contato e problemas com descontinuidades. Nesse sentido, sugere-se a investigação da influência do número de ciclos MEFG^{gl} (n_{GL}) em problemas fortemente não lineares. O enriquecimento da mesoescala altera a sua matriz de

rigidez. A diferença de rigidez entre a representação do domínio local na escala global e na mesoescala condiciona a convergência do GLI. Considerar avaliar o uso de relaxação dinâmica nesse tipo de problema.

3. Em complemento à sugestão anterior, simular a presença de trincas e, na sequência, a propagação dessas trincas. Sugere-se a aplicação da abordagem automática de determinação da região local do MEF G^{gl} proposta por Fonseca (2019).
4. Implementar a solução de mais de um problema local (e sua respectiva mesoescala) associado a um mesmo problema global. Neste caso, seria interessante que a implementação considerasse a computação paralela das soluções locais.
5. Avaliar outras técnicas para a aceleração da convergência do processo iterativo para obtenção da solução. Sugere-se a avaliação da técnica quasi-Newton, apresentada na Seção 3.3.3 e já implementada no código disponibilizado neste trabalho.
6. Estudar o uso conjunto de outras funções de enriquecimento além das provenientes da estratégia global-local na mesoescala. Há, nesse caso, preocupação especial em relação aos nós de interface entre a escala global e a mesoescala. Ainda que as malhas dessas escalas sejam coincidentes na interface, com o enriquecimento da mesoescala, outros graus de liberdade podem estar presentes a depender do enriquecimento utilizado.
7. Avaliar o efeito da zona *buffer*. Sugere-se que essa avaliação seja realizada em combinação com os modelos M1, M2 e M3, apresentados na Seção 4.3.3. Nesse caso, entretanto, pode-se variar a região enriquecida com a solução numérica do problema local, reduzindo-a, por exemplo, à região em que existe o fenômeno local de interesse.
8. Propor alterações no fluxo de solução original do MEF G^{gl} -GLI, apresentado na Fig. 2.11 e no Algoritmo 3.1. Por exemplo, os resultados obtidos neste trabalho indicam que os ciclos entre a mesoescala e a escala local, realizados pelo MEF G^{gl} , são onerosos. Nesse sentido, pode-se avaliar a necessidade da realização de ciclos MEF G^{gl} a cada iteração do GLI. Outra sugestão é modificar o tipo de condição de contorno transferida da escala global para a mesoescala.

Bibliografia

Aitken, A. C. (1927), ‘Xxv.—on bernoulli’s numerical solution of algebraic equations’, *Proceedings of the Royal Society of Edinburgh* vol. 46, 289–305.

URL: <http://dx.doi.org/10.1017/S0370164600022070>

Allix, O. e Gosselet, P. (2020), ‘Non intrusive global/local coupling techniques in solid mechanics: An introduction to different coupling strategies and acceleration techniques’, *Modeling in Engineering Using Innovative Numerical Methods for Solids and Fluids* p. 203–220.

URL: http://dx.doi.org/10.1007/978-3-030-37518-8_6

Alves, P. D. (2012), Estrat gia global-local aplicada ao m todo dos elementos finitos generalizados, Disserta o de mestrado, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, MG, Brasil.

Babuška, I., Caloz, G. e Osborn, J. E. (1994), ‘Special finite element methods for a class of second order elliptic problems with rough coefficients’, *SIAM Journal on Numerical Analysis* vol. 31, 945–981.

Barros, F. B. (2002), M todos Sem Malha e M todos dos Elementos Finitos Generalizados em An lise N o-Linear de Estruturas, Tese de doutorado, Universidade de S o Paulo (USP), S o Carlos, SP, Brasil.

Belytschko, T. e Black, T. (1999), ‘Elastic crack growth in finite elements with minimal remeshing’, *International Journal for Numerical Methods in Engineering* vol. 45(5), 601–620.

URL: [http://dx.doi.org/10.1002/\(SICI\)1097-0207\(19990620\)45:5<601::AID-NME598>3.0.CO;2-S](http://dx.doi.org/10.1002/(SICI)1097-0207(19990620)45:5<601::AID-NME598>3.0.CO;2-S)

Belytschko, T., Gracie, R. e Ventura, G. (2009), ‘A review of extended/generalized finite element methods for material modelling’, *Modelling and Simulation in Materials Science and Engineering* vol. 17, 043001.

URL: <http://dx.doi.org/10.1088/0965-0393/17/4/043001>

- Belytschko, T., Lu, Y. Y. e Gu, L. (1994), ‘Element-free galerkin methods’, *International Journal for Numerical Methods in Engineering* vol. 37, 229–256.
URL: <http://dx.doi.org/10.1002/nme.1620370205>
- Bernardi, C., Maday, Y. e Patera, A. T. (1989), ‘A new nonconforming approach to domain decomposition: the mortar element method’, *Brezis, H.(ed.) et al., Nonlinear partial differential equations and their applications. Collège de France Seminar, volume XI. Lectures presented at the weekly seminar on applied mathematics, Paris, France, .*
- Bernardi, C., Maday, Y. e Patera, A. T. (1993), ‘Domain decomposition by the mortar element method’, *Asymptotic and Numerical Methods for Partial Differential Equations with Critical Parameters* p. 269–286.
- Bettinotti, O., Allix, O., Perego, U., Oancea, V. e Malherbe, B. (2014), ‘A fast weakly intrusive multiscale method in explicit dynamics’, *International Journal for Numerical Methods in Engineering* vol. 100(8), 577–595.
URL: <http://dx.doi.org/10.1002/nme.4750>
- Bjørstad, P. E. e Widlund, O. B. (1986), ‘Iterative methods for the solution of elliptic problems on regions partitioned into substructures’, *SIAM Journal on Numerical Analysis* vol. 23(6), 1097–1120.
URL: <http://dx.doi.org/10.1137/0723075>
- Campos, B. C. (2020), Integração numérica por decomposição de elementos no contexto do método dos elementos finitos generalizados, Dissertação de mestrado, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, MG, Brasil.
- Chevreuril, M., Nouy, A. e Safatly, E. (2013), ‘A multiscale method with patch for the solution of stochastic partial differential equations with localized uncertainties’, *Computer Methods in Applied Mechanics and Engineering* vol. 255, 255–274.
URL: <http://dx.doi.org/10.1016/j.cma.2012.12.003>
- Duarte, C., Babuška, I. e Oden, J. (2000), ‘Generalized finite element methods for three-dimensional structural mechanics problems’, *Computers & Structures* vol. 77(2), 215–232.
URL: [http://dx.doi.org/10.1016/S0045-7949\(99\)00211-4](http://dx.doi.org/10.1016/S0045-7949(99)00211-4)
- Duarte, C. e Kim, D.-J. (2008), ‘Analysis and applications of a generalized finite element method with global–local enrichment functions’, *Computer Methods in Applied Mechanics and Engineering* vol. 197(6–8), 487–504.
URL: <http://dx.doi.org/10.1016/j.cma.2007.08.017>

- Duarte, C. e Oden, J. (1996), ‘An h-p adaptive method using clouds’, *Computer Methods in Applied Mechanics and Engineering* vol. 139(1–4), 237–262.
URL: [http://dx.doi.org/10.1016/S0045-7825\(96\)01085-7](http://dx.doi.org/10.1016/S0045-7825(96)01085-7)
- Duarte, C. e Oden, J. T. (1995a), Hp clouds – a meshless method to solve boundary-value problem, Technical Report TICAM Report 95-05 (Revised), Texas Institute for Computational and Applied Mathematics.
URL: <https://www.oden.utexas.edu/>
- Duarte, C. e Oden, J. T. (1995b), A review of some meshless methods to solve partial differential equations, Technical Report TICAM Report 95-06, Texas Institute for Computational and Applied Mathematics.
URL: <https://www.oden.utexas.edu/>
- Duval, M., Passieux, J.-C., Salaün, M. e Guinard, S. (2014), ‘Non-intrusive coupling: Recent advances and scalable nonlinear domain decomposition’, *Archives of Computational Methods in Engineering* vol. 23(1), 17–38.
URL: <http://dx.doi.org/10.1007/s11831-014-9132-x>
- Fillmore, T. B. e Duarte, C. A. (2018), ‘A hierarchical non-intrusive algorithm for the generalized finite element method’, *Advanced Modeling and Simulation in Engineering Sciences* vol. 5(1).
URL: <http://dx.doi.org/10.1186/s40323-018-0096-5>
- Fonseca, F. T. e Pitangueira, R. L. (2007), An object oriented class organization for dynamic geometrically non-linear, in ‘Iberian-Latin-American Congress on Computational Methods in Engineering (CILAMCE)’.
- Fonseca, G. M. (2019), Propagação de trincas em meios elásticos lineares via método dos elementos finitos generalizados com estratégia global-local automatizada, Dissertação de mestrado, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, MG, Brasil.
- Fuina, J. S. (2004), Métodos de controle de deformações para análise não-linear de estruturas, Dissertação de mestrado, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, MG, Brasil.
- Gander, M. J. e Japhet, C. (2013), ‘Algorithm 932’, *ACM Transactions on Mathematical Software* vol. 40(1), 1–25.
URL: <http://dx.doi.org/10.1145/2513109.2513115>
- Gendre, L., Allix, O. e Gosselet, P. (2011), ‘A two-scale approximation of the schur complement and its use for non-intrusive coupling’, *International Journal for Numerical*

Methods in Engineering vol. 87(9), 889–905.

URL: <http://dx.doi.org/10.1002/nme.3142>

Gendre, L., Allix, O., Gosselet, P. e Comte, F. (2009), ‘Non-intrusive and exact global/local techniques for structural problems with local plasticity’, *Computational Mechanics* vol. 44(2), 233–245.

URL: <http://dx.doi.org/10.1007/s00466-009-0372-9>

Gosselet, P. e Rey, C. (2006), ‘Non-overlapping domain decomposition methods in structural mechanics’, *Archives of Computational Methods in Engineering* vol. 13(4), 515–572.

URL: <http://dx.doi.org/10.1007/BF02905857>

Guguin, G., Allix, O., Gosselet, P. e Guinard, S. (2014), ‘Nonintrusive coupling of 3d and 2d laminated composite models based on finite element 3d recovery’, *International Journal for Numerical Methods in Engineering* vol. 98(5), 324–343.

URL: <http://dx.doi.org/10.1002/nme.4630>

Guidault, P.-A., Allix, O., Champaney, L. e Cornuault, C. (2008), ‘A multiscale extended finite element method for crack propagation’, *Computer Methods in Applied Mechanics and Engineering* vol. 197(5), 381–399.

URL: <http://dx.doi.org/10.1016/j.cma.2007.07.023>

Guinard, S., Bouclier, R., Toniolli, M. e Passieux, J.-C. (2018), ‘Multiscale analysis of complex aeronautical structures using robust non-intrusive coupling’.

URL: <http://dx.doi.org/10.1186/s40323-017-0094-z>

Gupta, P., Pereira, J., Kim, D.-J., Duarte, C. e Eason, T. (2012), ‘Analysis of three-dimensional fracture mechanics problems: A non-intrusive approach using a generalized finite element method’, *Engineering Fracture Mechanics* vol. 90, 41–64.

URL: <http://dx.doi.org/10.1016/j.engfracmech.2012.04.014>

Gupta, V., Kim, D.-J. e Duarte, C. A. (2012), ‘Analysis and improvements of global–local enrichments for the generalized finite element method’, *Computer Methods in Applied Mechanics and Engineering* vol. 245–246, 47–62.

URL: <http://dx.doi.org/10.1016/j.cma.2012.06.021>

Irons, B. M. e Tuck, R. C. (1969), ‘A version of the aitenk accelerator for computer iteration’, *International Journal for Numerical Methods in Engineering* vol. 1(3), 275–277.

URL: <http://dx.doi.org/10.1002/nme.1620010306>

- Kim, D.-J., Pereira, J. P. e Duarte, C. A. (2009), ‘Analysis of three-dimensional fracture mechanics problems: A two-scale approach using coarse-generalized fem meshes’, *International Journal for Numerical Methods in Engineering* vol. 81(3), 335–365.
URL: <http://dx.doi.org/10.1002/nme.2690>
- Küttler, U. e Wall, W. A. (2008), ‘Fixed-point fluid–structure interaction solvers with dynamic relaxation’, *Computational Mechanics* vol. 43(1), 61–72.
URL: <http://dx.doi.org/10.1007/s00466-008-0255-5>
- Ladevèze, P., Loiseau, O. e Dureisseix, D. (2001), ‘A micro–macro and parallel computational strategy for highly heterogeneous structures’, *International Journal for Numerical Methods in Engineering* vol. 52(12), 121–138.
URL: <http://dx.doi.org/10.1002/nme.274>
- Lancaster, P. e Salkauskas, K. (1981), ‘Surfaces generated by moving least squares methods’, *Mathematics of Computation* vol. 37(155), 141–158.
URL: <http://dx.doi.org/10.1090/S0025-5718-1981-0616367-1>
- Li, H., O’Hara, P. e Duarte, C. (2021), ‘Non-intrusive coupling of a 3-d generalized finite element method and abaqus for the multiscale analysis of localized defects and structural features’, *Finite Elements in Analysis and Design* vol. 193, 103554.
URL: <http://dx.doi.org/10.1016/j.finel.2021.103554>
- Liu, Y., Sun, Q. e Fan, X. (2014), ‘A non-intrusive global/local algorithm with non-matching interface: Derivation and numerical validation’, *Computer Methods in Applied Mechanics and Engineering* vol. 277, 81–103.
URL: <http://dx.doi.org/10.1016/j.cma.2014.04.012>
- Malekan, M. (2017), Crack propagation modeling in plane structures using two-scale generalized/extended finite element method, Tese de doutorado, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, MG, Brasil.
- Melenk, J. e Babuška, I. (1996), ‘The partition of unity finite element method: Basic theory and applications’, *Computer Methods in Applied Mechanics and Engineering* vol. 139(1–4), 289–314.
URL: [http://dx.doi.org/10.1016/S0045-7825\(96\)01087-0](http://dx.doi.org/10.1016/S0045-7825(96)01087-0)
- Moës, N., Dolbow, J. e Belytschko, T. (1990), ‘A finite element method for crack growth without remeshing’, *International Journal for Numerical Methods in Engineering* vol. 46(1), 131–150.
- Noor, A. K. (1986), ‘Global-local methodologies and their application to nonlinear analysis’, *Finite Elements in Analysis and Design* vol. 2(4), 333–346.
URL: [http://dx.doi.org/10.1016/0168-874X\(86\)90020-X](http://dx.doi.org/10.1016/0168-874X(86)90020-X)

- Oden, J. T., Duarte, C. e Zienkiewicz, C. (1996), A new cloud-based hp-finite element method, Technical Report TICAM Report 96-55, Texas Institute for Computational and Applied Mathematics.
URL: <https://www.oden.utexas.edu/>
- O'Hara, P., Duarte, C. e Eason, T. (2009), 'Generalized finite element analysis of three-dimensional heat transfer problems exhibiting sharp thermal gradients', *Computer Methods in Applied Mechanics and Engineering* vol. 198(21–26), 1857–1871.
URL: <http://dx.doi.org/10.1016/j.cma.2008.12.024>
- Penna, S. S. (2007), Pós-processador para modelos bidimensionais não-lineares do método dos elementos finitos, Dissertação de mestrado, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, MG, Brasil.
- Pinheiro, D. C. C. (2019), Estratégias baseadas na partição da unidade para simulação do comportamento de meios parcialmente frágeis, Tese de doutorado, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, MG, Brasil.
- Pitangueira, R. L. S., Fonseca, F. T., Fuina, J. S., Camara, L., Ferreira, R. L., Moreira, R. N., Penna, S. S., Saliba, S. S. e Fonseca, M. T. (2008), Insane - versão 2.0, *in* 'Iberian-Latin-American Congress on Computational Methods in Engineering (CILAMCE)'
- Plews, J., Duarte, C. e Eason, T. (2012), 'An improved nonintrusive global-local approach for sharp thermal gradients in a standard fea platform', *International Journal for Numerical Methods in Engineering* vol. 91(4), 426–449.
URL: <http://dx.doi.org/10.1002/nme.4279>
- Przemieniecki, J. S. (1963), 'Matrix structural analysis of substructures', *AIAA Journal* vol. 1(1), 138–147.
URL: <http://dx.doi.org/10.2514/3.1483>
- Shepard, D. (1968), 'A two-dimensional interpolation function for irregularly-spaced data', *Proceedings of the 1968 23rd ACM national conference on - .*
URL: <http://dx.doi.org/10.1145/800186.810616>
- Smith, M. (2016), *ABAQUS/Standard User's Manual, Version 6.14*, Dassault Systèmes Simulia Corp.
- Southwell, R. V. (1940), *Relaxation Methods in Engineering Science*, Oxford University Press.
- Southwell, R. V. (1946), *Relaxation Methods in Theoretical Physics*, Oxford University Press.

- Strouboulis, T., Babuška, I. e Copps, K. (2000), ‘The design and analysis of the generalized finite element method’, *Computer Methods in Applied Mechanics and Engineering* vol. 181(1–3), 43–69.
URL: [http://dx.doi.org/10.1016/S0045-7825\(99\)00072-9](http://dx.doi.org/10.1016/S0045-7825(99)00072-9)
- Tirvaudey, M., Chamoin, L., Bouclier, R. e Passieux, J.-C. (2020), ‘A posteriori error estimation and adaptivity in non-intrusive couplings between concurrent models’, *Computer Methods in Applied Mechanics and Engineering* vol. 367, 113104.
URL: <http://dx.doi.org/10.1016/j.cma.2020.113104>
- Whitcomb, J. (1991), ‘Iterative global/local finite element analysis’, *Computers & Structures* vol. 40(4), 1027–1031.
URL: [http://dx.doi.org/10.1016/0045-7949\(91\)90334-I](http://dx.doi.org/10.1016/0045-7949(91)90334-I)
- Wohlmuth, B. I. (2000), ‘A mortar finite element method using dual spaces for the lagrange multiplier’, *SIAM Journal on Numerical Analysis* vol. 38(3), 989–1012.
URL: <http://dx.doi.org/10.1137/S0036142999350929>
- Wyart, E., Duflot, M., Coulon, D., Martiny, P., Pardoën, T., Remacle, J.-F. e Lani, F. (2008), ‘Substructuring fe–xfe approaches applied to three-dimensional crack propagation’, *Journal of Computational and Applied Mathematics* vol. 215(2), 626–638.
URL: <http://dx.doi.org/10.1016/j.cam.2006.03.066>
- Xu, J. (1992), ‘Iterative methods by space decomposition and subspace correction’, *SIAM Review* vol. 34(4), 581–613.
URL: <http://dx.doi.org/10.1137/1034116>