

UNIVERSIDADE FEDERAL DE MINAS GERAIS
Escola de Engenharia
Programa de Pós-Graduação em Engenharia de Estruturas

Ana Clara Pedras Bueno

**IMPLEMENTAÇÃO NÃO INTRUSIVA DO MÉTODO DOS
ELEMENTOS FINITOS GENERALIZADOS
GLOBAL-LOCAL PARA SIMULAÇÃO DE PROBLEMAS
COM TRINCAS**

Belo Horizonte
2024

Ana Clara Pedras Bueno

**IMPLEMENTAÇÃO NÃO INTRUSIVA DO MÉTODO DOS ELEMENTOS
FINITOS GENERALIZADOS GLOBAL-LOCAL PARA SIMULAÇÃO DE
PROBLEMAS COM TRINCAS**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Estruturas da Escola de Engenharia da Universidade Federal de Minas Gerais, como parte do requisitos necessários à obtenção do título de Mestre em Engenharia de Estruturas

Orientador: Prof. Dr. Felício Bruzzi Barros

Belo Horizonte
2024

B928i	<p>Bueno, Ana Clara Pedras. Implementação não intrusiva do método dos elementos finitos generalizados global-local para simulação de problemas com trincas [recurso eletrônico] / Ana Clara Pedras Bueno. - 2024. 1 recurso online (121 f. : il., color.) : pdf.</p> <p>Orientador: Felício Bruzzi Barros.</p> <p>Dissertação (mestrado) - Universidade Federal de Minas Gerais, Escola de Engenharia.</p> <p>Apêndices: f. 116-121.</p> <p>Bibliografia: f. 111-115.</p> <p>1. Engenharia de estruturas - Teses. 2. Mecânica da fratura - Teses. 3. Método dos elementos finitos - Teses. 4. Acoplamentos - Teses. I. Barros, Felício Bruzzi. II. Universidade Federal de Minas Gerais. Escola de Engenharia. III. Título.</p> <p style="text-align: right;">CDU: 624(043)</p>
-------	---



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: 406 DA ALUNA ANA CLARA PEDRAS BUENO

Às **13:00** horas do dia **28** do mês de **junho** de **2024**, reuniu-se, na sala 4409 da Escola de Engenharia, com a participação remota do examinador externo, a Comissão Examinadora a saber, indicada pelo Colegiado do Programa em **14 de junho de 2024**, para julgar a defesa da Dissertação de Mestrado intitulada "**Implementação não Intrusiva do Método dos Elementos Finitos Generalizados Global-local para Simulação de Problemas com Trincas**", cuja aprovação é um dos requisitos para a obtenção do Grau de MESTRE EM ENGENHARIA 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 do Programa, passou a palavra à candidata para apresentação de seu trabalho. Seguiu-se a arguição dos examinadores, com a respectiva defesa da candidata. Logo após, a Comissão se reuniu, sem a presença da candidata 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. Eduardo de Moraes Barreto Campello - Poli - USP

Pelas indicações acima, à candidata foi considerada APROVADA, conforme pareceres em anexo.

O resultado final foi comunicado publicamente à candidata 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, 28 de junho de 2024.

Observações:

1. A aprovação da candidata na defesa da Dissertação de Mestrado não significa que a mesma 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-Graduação.



Documento assinado eletronicamente por **Felicio Bruzzi Barros, Professor(a)**, em 28/06/2024, às 16:09, 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 28/06/2024, às 16:11, 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 **Eduardo de Moraes Barreto Campello, Usuário Externo**, em 28/06/2024, às 16:12, 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 10/07/2024, às 09:58, 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 **3308964** e o código CRC **52A7AD13**.

À minha amada vó Dora.

AGRADECIMENTOS

A Deus, por me dar saúde e as oportunidades para concretizar os meus objetivos.

À minha família, em especial, aos meus pais Cássio e Soraia, e à minha irmã, Ana Vitória, que são a minha maior motivação.

Ao meu orientador, professor Felício, que ao longo de todo o trabalho se manteve disponível, atencioso e dedicado. E que se tornou um exemplo profissional, neste meu primeiro passo rumo à carreira acadêmica.

Aos demais professores do Departamento de Engenharia de Estruturas da UFMG, e à todos os colegas do grupo de pesquisa INSANE, por todos os ensinamentos compartilhados.

Aos meus amigos, por estarem e dividirem comigo cada momento desta jornada. Especialmente, à minha amiga Maria Tereza, e ao seus queridos padrinhos Betânia e Moca, que me receberam e me acolheram com todo amor e cuidado durante a minha temporada em Belo Horizonte.

À CAPES, pelo apoio financeiro.

RESUMO

Esta dissertação de mestrado expande a implementação da estratégia de acoplamento não intrusivo entre o *software* comercial Abaqus e o ambiente computacional INSANE (*INterative Structural ANalysis Environment*), através da estratégia MEFG^{gl}-GLI, dando ênfase ao tratamento de problemas da Mecânica da Fratura Linear Elástica (MFLE). A estratégia MEFG^{gl}-GLI combina o algoritmo Global-Local Iterativo (GLI) com o Método dos Elementos Finitos Generalizados com enriquecimento global-local (MEFG^{gl}), dividindo a análise em três escalas (global, meso e local) e seus respectivos modelos. Na implementação proposta, o modelo da escala global é solucionado pelo *software* comercial Abaqus via Método dos Elementos Finitos (MEF), enquanto o MEFG^{gl}, implementado no ambiente INSANE, realiza o enriquecimento do modelo da mesoescala a partir da solução do modelo da escala local. O acoplamento com a escala global é realizado iterativamente com a transferência de deslocamentos entre a escala global e a mesoescala e correção do resíduo de força, utilizando o algoritmo GLI. Através da estrutura não intrusiva do GLI, um algoritmo iterativo, juntamente com as técnicas de relaxação, foram implementadas no INSANE para acoplar o modelo da escala global com o modelo da mesoescala. A comunicação do INSANE com o Abaqus é realizada por intermédio de *scripts* desenvolvidos em linguagem de programação Python, e interpretados com auxílio do projeto Jython. A expansão proposta permitiu a descrição e propagação de trinca na escala local, e implementou uma abordagem computacionalmente mais eficiente da estratégia. A validação da implementação foi realizada a partir de simulações numéricas que investigaram parâmetros relacionados ao MEFG^{gl} dentro da abordagem da estratégia iterativa do acoplamento não intrusivo. As investigações revelaram que o aumento do tamanho da zona *buffer* da mesoescala melhora a aproximação das soluções das estratégias de acoplamento em relação a solução de referência. Observou-se também que não há melhora significativa na precisão dos resultados com o aumento do número de ciclos global-local, tal fato reside na compensação advinda da repetição do processo de solução global-local exigida para a convergência da estratégia iterativa do acoplamento. Verificou-se ainda, que a utilização da técnica de relaxação dinâmica acelera a convergência do procedimento iterativo sem comprometer a precisão dos resultados, contribuindo para que as estratégias MEFG^{gl}-GLI se tornem computacionalmente competitivas quando comparadas aos procedimentos de soluções convencionais.

Palavras-chave: acoplamento não intrusivo; Global-Local Iterativo; Método dos Elementos Finitos Generalizados com enriquecimento global-local; Mecânica da Fratura Linear Elástica.

ABSTRACT

This master's thesis expands the implementation of the non-intrusive coupling strategy between the commercial software Abaqus and the INSANE (INterative Structural ANalysis Environment) computational environment, through the IGL-GFEM^{gl} strategy, emphasizing the treatment of Linear Elastic Fracture Mechanics (LEFM) problems. The IGL-GFEM^{gl} strategy combines the Iterative Global-Local (IGL) algorithm with the Generalized Finite Element Method with global-local enrichment (GFEM^{gl}), dividing the analysis into three scales (a global, a meso and a local scale) and their respective models. In the implementation proposed, the global scale model is solved by the commercial software Abaqus via the Finite Element Method (FEM), while GFEM^{gl}, implemented in the INSANE environment, performs the enrichment of the mesoscale model from the solution of the local scale model. The coupling with the global scale is iteratively provided by transferring displacements between the global scale and the mesoscale and the correction of the residual forces, using the IGL algorithm. Through the non-intrusive procedure of IGL, an iterative algorithm, and relaxation techniques were implemented in INSANE to couple the global scale model with the mesoscale model. INSANE's communication with Abaqus is carried out through scripts developed in Python programming language and interpreted with the help of the Jython project. The proposed expansion allowed the description and propagation of cracks at the local scale and implemented a more computationally efficient approach to the strategy. The implementation validation was carried out based on numerical simulations investigating parameters related to GFEM^{gl} under the interactive strategy of non-intrusive coupling. The investigations revealed that the increase in the size of the buffer zone of the mesoscale improves the solution when compared with a reference model. It was also observed that there was no significant improvement in the accuracy of the results with the increase in the number of global-local cycles, such a fact can be explained by the repetition of the global-local solution process required for converging the iterative coupling strategy. It was also verified that the dynamic relaxation technique accelerates the convergence of the coupling strategies without compromising the accuracy of the results. This technique contributed to making IGL-GFEM^{gl} comparatively competitive with standard solution procedures.

Key-words: non-intrusive coupling; Iterative Global-Local; Generalized Finite Element Method with global-local enrichment; Linear Elastic Fracture Mechanics.

LISTA DE FIGURAS

2.1	Técnica <i>Zooming</i> : Modelo global com malha grosseira é utilizado para determinar as condições de contorno a serem aplicadas ao longo da interface do modelo local refinado.	27
2.2	Ilustração do processo de solução do MEF Global-Local Iterativo (MEF-GLI) (adaptado de Silveira Filho (2023)).	29
2.3	Estratégia de enriquecimento do MEFGL (adaptado de Barros (2002)).	35
2.4	Domínio em \mathbb{R}^2 com vértice V (adaptado de Szabo e Babuška (1991)).	37
2.5	Representação esquemática do enriquecimento global-local para um problema bidimensional. Os nós globais enriquecidos estão destacados em amarelo.	39
2.6	Representação esquemática do domínio global e local no MEFGL para um problema bidimensional.	40
2.7	Processo de ciclo global-local no qual as condições de contorno para o problema local do ciclo $j + 1$ são obtidas da solução do problema global enriquecido do ciclo j (adaptado de Fonseca (2019)).	43
2.8	Representação esquemática da zona <i>buffer</i>	44
2.9	Representação esquemática da abordagem da estratégia MEFGL-GLI em um pórtico em “L” com a inclusão de uma trinca.	46
2.10	Fluxograma dos principais passos da estratégia MEFGL-GLI segundo abordagem monolítica.	48
2.11	Fluxograma dos principais passos da estratégia MEFGL-GLI segundo abordagem do algoritmo <i>staggered</i>	50
3.1	Diagrama de sequência dos principais passos do procedimento de solução da abordagem monolítica.	58
3.2	Diagrama de sequência dos principais passos do procedimento de solução da abordagem <i>staggered</i>	59
3.3	Notação UML utilizada nos diagramas de classes.	60
3.4	Modificações da classe <code>PersistenceAsXml</code>	61
3.5	Criação da classe <code>IGLGlobalLocal</code>	62
3.6	Criação da interface <code>IglAcceleration</code>	64
3.7	Criação da classe <code>IGLGlobalLocalCrackPropagationSimulationManager</code>	65

3.8	Modificação da classe <code>LocalSetuperGfem</code>	67
4.1	Chapa tracionada com orifício quadrado.	69
4.2	Deslocamentos na direção y obtidos através do $\text{MEFG}^{gl}\text{-GLI}^{St}$	70
4.3	Deslocamentos na direção y obtidos pelo Modelo A.	71
4.4	Deslocamentos na direção y obtidos pelo Modelo B.	71
4.5	Comparação dos resultados em termos de u_y ao longo da borda inferior da chapa obtidos pelo $\text{MEFG}^{gl}\text{-GLI}^{St}$, pelo $\text{MEFG}^{gl}\text{-GLI}^{M1}$, e pelos modelos A e B.	72
4.6	Tensões na direção x obtidos através do $\text{MEFG}^{gl}\text{-GLI}^{St}$, e pelos modelos A e B.	73
4.7	Definição dos pontos para comparação dos resultados da chapa.	74
4.8	Comparação do comportamento da convergência das estratégias $\text{MEFG}^{gl}\text{-GLI}^{St}$ e $\text{MEFG}^{gl}\text{-GLI}^{M2}$ com $n_{GL} = 1$	76
4.9	Comparação do comportamento da convergência das estratégias $\text{MEFG}^{gl}\text{-GLI}^{St}$ e $\text{MEFG}^{gl}\text{-GLI}^{M2}$ com $n_{GL} = 2$	77
4.10	Identificação dos nós enriquecidos com funções de singularidade.	78
4.11	Comparação do comportamento da convergência das estratégias $\text{MEFG}^{gl}\text{-GLI}^{St}$ e $\text{MEFG}^{gl}\text{-GLI}^{M2}$ com enriquecimento do modelo da escala local e $n_{GL} = 2$	78
4.12	Comparação dos resultados em termos de u_y ao longo da borda inferior da chapa obtidos pelo $\text{MEFG}^{gl}\text{-GLI}^{St}$ e pelo $\text{MEFG}^{gl}\text{-GLI}^{M2}$ com enriquecimento do modelo da escala local, e pelos modelos A e B.	79
4.13	Pórtico em “L” com a inclusão de uma trinca.	80
4.14	Problema global enriquecido no 1° e no 7° passo de propagação com ausência de zona <i>buffer</i>	83
4.15	Comportamento da convergência de K_I e K_{II} em função do número de ciclos global-local no 1° e no 7° passo de propagação assumindo domínio local mínimo.	83
4.16	Problema global enriquecido no 1° e no 7° passo de propagação com zona <i>buffer</i> de uma camada de elementos.	85
4.17	Problema global enriquecido no 1° e no 7° passo de propagação com zona <i>buffer</i> de duas camada de elementos.	86
4.18	Comportamento da convergência de K_I em função do número de ciclos global-local e tamanho da zona <i>buffer</i> no 7° passo de propagação.	87
4.19	Comportamento da convergência de K_{II} em função do número de ciclos global-local e tamanho da zona <i>buffer</i> no 7° passo de propagação.	87
4.20	Passos de propagação da trinca obtidos pela solução de referência no modelo equivalente.	88

4.21	Resultados da solução de referência do modelo numérico equivalente em termos de u_y	94
4.22	Comparação da eficiência computacional das estratégias $MEFG^{gl}$ - GLI^{St} e $MEFG^{gl}$ - GLI^{M2} , sem e com relaxação dinâmica, em função do número de ciclos global-local.	98
A.1	Pasta de trabalho com os arquivos de entrada.	117
B.1	Arquivo de entrada <code>meso.xml</code> para definição da solução <code>IGLGlobalLocal</code>	119
B.2	Identificação dos nós e dos elementos de interface.	120

LISTA DE TABELAS

3.1	Principais atributos da classe.	66
4.1	Comparação dos resultados em termos de σ_{xx} obtidos pelo MEFG ^{gl} -GLI St , e pelos modelos A e B.	74
4.2	Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente no 1° passo de propagação da trinca.	82
4.3	Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente no 7° passo de propagação da trinca.	82
4.4	Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente com zona <i>buffer</i> de uma camada de elementos no 1° passo de propagação da trinca.	84
4.5	Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente com zona <i>buffer</i> de uma camada de elementos no 7° passo de propagação da trinca.	84
4.6	Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente com zona <i>buffer</i> de duas camadas de elementos no 1° passo de propagação da trinca.	86
4.7	Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente com zona <i>buffer</i> de duas camadas de elementos no 7° passo de propagação da trinca.	86
4.8	Resultados em termos dos fatores de intensidade de tensão obtidos pela solução de referência e pela estratégia MEFG ^{gl} -GLI ^{M2}	92
4.9	Resultados em termos dos fatores de intensidade de tensão obtidos pela solução de referência e pela estratégia MEFG ^{gl} -GLI St	92
4.10	Resultados em termos dos fatores de intensidade de tensão obtidos pela solução de referência e pela estratégia MEFG ^{gl} -GLI ^{M2} adotando relaxação dinâmica.	93
4.11	Resultados em termos dos fatores de intensidade de tensão obtidos pela solução de referência e pela estratégia MEFG ^{gl} -GLI St adotando relaxação dinâmica.	93

4.12	Comparação dos resultados em termos de u_y ao longo da extremidade direita do pórtico, obtidos pela solução de referência do modelo numérico equivalente e pelas estratégias MEFG ^{gl} -GLI adotando relaxação dinâmica.	94
4.13	Comparação da eficiência computacional da estratégia MEFG ^{gl} -GLI ^{M2} sem relaxação dinâmica.	96
4.14	Comparação da eficiência computacional da estratégia MEFG ^{gl} -GLI ^{M2} com relaxação dinâmica.	96
4.15	Comparação da eficiência computacional da estratégia MEFG ^{gl} -GLI St sem relaxação dinâmica.	97
4.16	Comparação da eficiência computacional da estratégia MEFG ^{gl} -GLI St com relaxação dinâmica.	97
4.17	Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente e pela estratégia MEFG ^{gl} -GLI ^{M2} adotando relaxação dinâmica e diferentes camadas de zona <i>buffer</i> .	101
4.18	Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente e pela estratégia MEFG ^{gl} -GLI St adotando relaxação dinâmica e diferentes camadas de zona <i>buffer</i> .	101
4.19	Comparação dos resultados em termos de u_y ao longo da extremidade direita do pórtico, obtidos pela solução de referência do modelo numérico equivalente e pelas estratégias MEFG ^{gl} -GLI adotando relaxação dinâmica e diferentes camadas de zona <i>buffer</i> .	102
4.20	Comparação da eficiência computacional da estratégia MEFG ^{gl} -GLI ^{M2} adotando relaxação dinâmica e diferentes camadas de zona <i>buffer</i> .	104
4.21	Comparação da eficiência computacional da estratégia MEFG ^{gl} -GLI St adotando relaxação dinâmica e diferentes camadas de zona <i>buffer</i> .	104
4.22	Comparação da eficiência computacional das estratégias MEFG ^{gl} -GLI ^{M2} e MEFG ^{gl} -GLI St , adotando novo modelo da escala local, duas camadas de zona <i>buffer</i> , e relaxação dinâmica.	106

LISTA DE ABREVIACOES

EPD	Estado Plano de Deformao
EPT	Estado Plano de Tenso
GLI	Global-Local Iterativo
HNA	<i>Hierarchical Non-Intrusive Algorithm</i>
IA	Inteligncia Artificial
IGL-FEM	<i>Iterative Global-Local Finite Element Method</i>
INSANE	<i>INteractive Structural ANalysis Environment</i>
MEC	Mtodo dos Elementos de Contorno
MEF	Mtodo dos Elementos Finitos
MEF-GL	Mtodo dos Elementos Finitos Global-Local
MEF-GLI	Mtodo dos Elementos Finitos Global-Local Iterativo
MEFE	Mtodo dos Elementos Finitos Estendido
MEFG	Mtodo dos Elementos Finitos Generalizados
MEFG ^{gl}	Mtodo dos Elementos Finitos Generalizados com enriquecimento Global-Local
MEFG ^{gl} -GLI	Estratgia Global-Local Iterativa aplicada ao MEFG ^{gl}
MEFG ^{gl} -GLI ^{M1}	Abordagem Monoltica da estratgia MEFG ^{gl} -GLI proposta por Li <i>et al.</i> (2021) e implementada por Silveira Filho (2023)
MEFG ^{gl} -GLI ^{M2}	Abordagem Monoltica da estratgia MEFG ^{gl} -GLI proposta no presente trabalho
MEFG ^{gl} -GLI St	Abordagem <i>Staggered</i> da estratgia MEFG ^{gl} -GLI
MM	Mtodos sem Malha
MFLE	Mecnica da Fratura Linear Elstica
POO	Programao Orientada  Objetos
PU	Partio da Unidade
UML	<i>Unified Modelling Language</i>
XFEM	<i>eXtended Finite Element Method</i>

LISTA DE SÍMBOLOS

Sobrescritos

\cdot^0	Índice que se refere aos graus de liberdade do problema global inicial
$\cdot^{(1)}$	Índice relativo ao modo I de abertura
$\cdot^{(2)}$	Índice relativo ao modo II de abertura
\cdot^E	Índice que se refere ao problema global enriquecido
\cdot^D	Índice relativo às condições de contorno de Dirichlet
\cdot^{M1}	Índice relativo à estratégia de acoplamento monolítico proposta por Li <i>et al.</i> (2021) e implementada por Silveira Filho (2023)
\cdot^{M2}	Índice relativo à estratégia de acoplamento monolítico proposta no presente trabalho
\cdot^N	Índice relativo às condições de contorno de Neumann
\cdot^{Ref}	Índice relativo à solução de referência
\cdot^{St}	Índice relativo à estratégia de acoplamento <i>staggered</i>
\cdot^i	Índice relativo à iteração do GLI
\cdot^j	Índice relativo à iteração do MEF G^{gl}
\cdot^{gl}	Índice que se refere aos graus de liberdade introduzidos pelo enriquecimento
\cdot^{Γ_I}	Índice relativo à interface entre domínios

Subscritos

\cdot_C	Índice relativo ao domínio complementar
\cdot_G	Índice relativo ao domínio global
\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
\cdot_I	Índice relativo à interface
\cdot_M	Índice relativo ao domínio da mesoescala
\cdot_{MEnr}	Índice relativo ao domínio da mesoescala enriquecido
\cdot_R	Índice relativo às forças residuais
\cdot_{gl}	Índice relativo à iteração global-local
\cdot_{gli}	Índice relativo à iteração GLI
\cdot_s	Índice relativo à iteração <i>staggered</i>
\cdot_{tot}	Índice relativo às iterações totais

Letras maiúsculas

E	Módulo de elasticidade longitudinal
G	Módulo de elasticidade transversal
H	Funções de Heaviside
\mathcal{H}^1	Espaço de Hilbert de ordem 1
I_j	Conjunto de funções de enriquecimento do MEFG
J	Jacobiano do elemento global
\mathbf{K}	Matriz de rigidez
L_{ji}	Funções de enriquecimento do MEFG
\mathcal{N}_j	Funções Lagrangianas Lineares
Q	Constante da função de singularidade
V	Volume do elemento utilizado no sistema paramétrico
\mathcal{X}	Espaço de discretização

Letras minúsculas

a	comprimento inicial da trinca
\mathbf{b}_{ji}	Parâmetros nodais associados ao MEFG
\mathbf{f}	Vetor de forças
l	dimensão do elemento finito
n	Número de dimensões espaciais
n_{GL}	Número de ciclos global-local
r	Coordenada Polar com origem na ponta da trinca
\mathbf{t}	Vetor de forças de superfície
$\bar{\mathbf{t}}$	Vetor de forças de superfície prescritas
u_y	Componente de deslocamento na direção y
\mathbf{u}	Solução numérica aproximada
$\tilde{\mathbf{u}}$	Solução genérica do MEFG
$\bar{\mathbf{u}}$	Solução global prevista
\mathbf{u}_G^0	Solução global inicial
\mathbf{u}_j	Parâmetros nodais associados ao MEF
\mathbf{u}_f	Solução numérica de referência do GLI
u_x	Função de singularidade na direção "x"
u_y	Função de singularidade na direção "y"
\mathbf{v}	Funções tentativa
x	Posição em um domínio
\mathbf{x}_j	Vetor de coordenadas de um nó j

Letras gregas maiúsculas

β	Inclinação inicial da trinca
Γ_I	Interface entre os domínios
Δ^i	Vetor de incremento de deslocamentos de Aitken
Δa	Incremento de propagação da trinca
Δu	Incremento da solução numérica
Ω	Domínio aberto e suave
$\partial\Omega$	Contorno do domínio Ω
ϕ_{ji}	Função de forma do MEFG

Letras gregas minúsculas

ϵ	Tolerância para o resíduo relativo
ε	Tensor de deformações
θ	Coordenada polar com origem na ponta da trinca
κ	Parâmetro de penalidade
$\bar{\kappa}$	Constante determinada a partir do coeficiente de Poisson
λ	Constante da função de singularidade
ν	Coefficiente de Poisson
ξ	Posição da descontinuidade
σ_{xx}	Componente de tensão na direção x
$\boldsymbol{\sigma}$	Tensor de tensões
ω	Fator de relaxação
ω_j	Nuvem de elementos que concorrem no nó j

Operadores e símbolos especiais

\forall	Para qualquer que seja
\cup	União de conjuntos
\cap	Interseção de conjuntos
\subset	Está contido
\emptyset	Conjunto vazio
$\ a\ $	Norma euclidiana do vetor \mathbf{a}
$\underline{\mathbf{a}}$	Prolongamento do vetor ou da matriz \mathbf{a}

Conjuntos numéricos

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

SUMÁRIO

1	Introdução	21
1.1	Contextualização	21
1.2	Justificativa	23
1.3	Objetivos	24
1.3.1	Objetivo geral	24
1.3.2	Objetivos específicos	24
1.4	Organização do texto	25
2	Revisão da Literatura	26
2.1	Método dos Elementos Finitos Global-Local	26
2.2	Estratégia não intrusiva MEF-GLI	28
2.2.1	Convergência do GLI	32
2.3	Método dos Elementos Finitos Generalizados	34
2.3.1	Formulação do MEF ^g	35
2.3.2	Modelagem de Trincas através do MEF ^g	36
2.3.3	Formulação do MEF ^{gl}	38
2.3.4	Ciclos global-local e Zona <i>buffer</i>	42
2.4	Estratégias de acoplamento para o MEF ^g	43
2.5	Estratégia não intrusiva MEF ^{gl} -GLI	46
3	Aspectos Computacionais	52
3.1	Procedimentos de solução do MEF ^{gl} -GLI	52
3.1.1	Algoritmo de solução monolítica	53
3.1.2	Algoritmo de solução <i>staggered</i>	55
3.2	Implementação Computacional	56
3.2.1	Entrada de dados	60
3.2.2	Classe IGLGlobalLocal	61
3.2.3	Classe IglAcceleration	63
3.2.4	Classe IGLGlobalLocalSimulationManager	64
4	Simulações Numéricas	68
4.1	Chapa com orifício quadrado	68

4.1.1	Deslocamento na direção y , u_y	70
4.1.2	Componente de tensão normal na direção x , σ_{xx}	72
4.1.3	Desempenho Computacional	75
4.2	Pórtico em “L” com propagação de trinca	80
4.2.1	Solução de referência no modelo equivalente	81
4.2.2	Influência do número de ciclos global-local	89
4.2.3	Influência do tamanho da zona <i>buffer</i>	99
5	Considerações Finais	107
5.1	Considerações deste Trabalho	107
5.2	Sugestões para Trabalhos Futuros	110
	Apêndices	116
A	Requisitos	117
B	Dados de Entrada	119

CAPÍTULO 1

INTRODUÇÃO

1.1 Contextualização

Uma variedade de problemas de engenharia podem ser descritos por modelos matemáticos baseados em equações diferenciais a partir de hipóteses simplificadoras. As soluções de tais equações diferenciais, ditas soluções analíticas, podem ser de complexa obtenção e conhecidas apenas para um número restrito de casos. Para superar as limitações destas soluções, métodos numéricos são utilizados para fornecer soluções aproximadas capazes de reproduzir o fenômeno físico de interesse.

As possibilidades oferecidas pelos recursos tecnológicos no desenvolvimento de *software* formam um vasto campo para pesquisas na área de métodos numéricos e computacionais aplicáveis a problemas de engenharia. Neste contexto, foi desenvolvido no Departamento de Engenharia de Estruturas da Universidade Federal de Minas Gerais o ambiente computacional INSANE (*INteractive Structural ANalysis Environment*). O ambiente INSANE é um projeto de *software* livre, implementado em linguagem de programação Java segundo o paradigma de Programação Orientada à Objetos (POO) (Pitangueira *et al.*, 2008). O projeto, que nasceu em 2002, foi concebido inicialmente como um sistema computacional para o Método dos Elementos Finitos (MEF), contudo, o desenvolvimento e o consequente amadurecimento do código possibilitou sua expansão para diferentes áreas e métodos numéricos (Pitangueira e Fonseca, 2007).

O INSANE configura hoje um ambiente computacional segmentado, amigável a mudanças e escalável em complexidade (Silva, 2016), e, propiciou, por isso, o desenvolvimento de diversos trabalhos acadêmicos em nível de graduação e pós-graduação. Dentre as principais linhas de pesquisa e trabalhos desenvolvidos no ambiente INSANE pode-se citar: Método dos Elementos Finitos (MEF), Método dos Elementos de Cotorno (MEC) (Peixoto *et al.*, 2018), Métodos sem Malha (MM) ((Gori *et al.*, 2018) e (Pinheiro *et al.*, 2022)), Método dos Elementos Finitos Generalizados (MEFG) (Alves *et al.*, 2013), aplicações em problemas de propagação de trinca em meios elásticos, e parcialmente frágeis ((Malekan *et al.*, 2018b) e (Wolff *et al.*, 2010)), Método dos Elementos Finitos Generalizados Global-Local (MEFG^{gl})((Alves, 2012) e (Malekan *et al.*, 2018a)), modelagem de concreto ((Gori *et al.*, 2019); (Oliveira *et al.*, 2020) e (Monteiro *et al.*, 2020)), problemas dinâmicos, mul-

tíficos (Botelho *et al.*, 2018) e investigação com Inteligência Artificial (IA) (Figueiredo *et al.*, 2020).

Os trabalhos desenvolvidos no ambiente INSANE envolvem, em geral, métodos e formulações que, embora bem consolidados em meio acadêmico, ainda não estão disponíveis em plenitude para a indústria. A implementação de tais métodos e formulações em pacotes de programas comerciais costuma ser difícil, demorada e, portanto, cara (Filmore e Duarte, 2018). Ao mesmo tempo, os programas de pesquisa que implementam tais métodos, e que poderiam fortalecer a indústria de *software* nacional, apresentam limitações no escopo de aplicação, por não priorizarem a eficiência computacional. Uma estratégia para resolver tal questão, é combinar a robustez do *software* comercial com os recursos do *software* de pesquisa para fornecer ao usuário final métodos de simulação e modelagem ainda não disponíveis em um único *software*.

Na prática, a combinação de simulações computacionais pode ser realizada a partir de métodos de acoplamento. De acordo com Duval *et al.* (2014), os métodos de acoplamento mais comuns utilizam procedimentos iterativos e/ou recursivos, muitas vezes combinados com estratégias de manipulação matricial. Contudo, tais métodos requerem alterações nas matrizes de rigidez e/ou nos *solvers* (solucionadores) envolvidos, o que nem sempre é possível em pacotes de programas comerciais. Esses métodos constituem as estratégias de acoplamento intrusivo.

Nos últimos anos, ganharam evidência diversos métodos de acoplamento não intrusivo (Duval *et al.*, 2014). Estes métodos dispensam alterações de *software* ou das estratégias de solução envolvidas. Assim, uma das principais consequências da não intrusividade é a possibilidade de combinar programas comerciais e de pesquisa sem alterações no código-fonte de nenhuma das aplicações. Entre as propostas de acoplamento não intrusivo destacam-se: acoplamento entre modelos 2D globais e 3D locais (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). Destacam-se, ainda, os trabalhos de Filmore e Duarte (2018) sobre algoritmos de acoplamento hierárquico não intrusivo para o MEF G^l , Li *et al.* (2021) sobre implementação não intrusiva do MEF G^l para a análise de problemas multiescala (MEF G^l -GLI), e seu desdobramento em Li *et al.* (2022) para o acoplamento multiescala de modelos sólidos e de casca 3D.

Recentemente, Silveira Filho (2023) realizou a implementação não intrusiva para a análise de problemas multiescala através da estratégia MEF G^l -GLI em sua abordagem monolítica, conforme proposto por Li *et al.* (2021). As simulações realizadas pelo referido autor restringiram-se à classe de problemas com comportamento linear elástico com a inclusão de vazios ou a variação da rigidez do material na escala local. As informações da escala local são levadas para a escala intermediária sob forma de funções de enriquecimento obtidas numericamente em tempo de execução, empregando-se o MEF G^l no ambiente IN-

SANE. A escala global é solucionada pelo *software* comercial *Abaqus* (Abaqus, 2014) via MEF. O acoplamento entre a escala global e a escala intermediária é realizado iterativamente por uma aplicação desenvolvida em linguagem de programação Python (Python, 2021), que gerencia a transferência de deslocamentos e correção de resíduo de força entre as escalas, conforme a estratégia Global-Local Iterativa (GLI) proposta por Whitcomb (1991).

O presente trabalho expande o desenvolvimento de Silveira Filho (2023), dando ênfase ao tratamento de problemas da Mecânica da Fratura Linear Elástica (MFLE), permitindo, portanto, a descrição e propagação de trinca na escala local. Tal expansão implementa uma nova abordagem da estratégia não intrusiva do MEF G^{gl} -GLI, conforme proposto por Li *et al.* (2022), e utiliza a contribuição de Fonseca (2019) no estudo de propagação de trincas em análise linear elástica através do MEF G^{gl} para solução da escala intermediária e da escala local.

1.2 Justificativa

A indústria, os laboratórios e a academia enfrentam um número crescente de desafios para modelagem e simulação de problemas de engenharia cada vez mais complexos, que envolvem, muitas vezes, interações entre múltiplas escalas espaciais. Em alguns casos, pode não ser viável utilizar uma única ferramenta de simulação que possua todos os recursos necessários para a resolução do problema com precisão.

Estruturas de engenharia na presença de fenômenos físicos localizados, tais como descontinuidades, trincas e não linearidades localizadas, por exemplo, requerem técnicas de modelagem avançadas e malhas refinadas para capturar de forma precisa a resposta em escala local. Contudo, adotar tais técnicas de discretização em escala global pode ser computacionalmente ineficiente, e até mesmo inviável para problemas que envolvem grande número de características locais de interesse (Li *et al.*, 2021). Por outro lado, uma discretização grosseira poderia ser suficiente para prever o comportamento da estrutura em escala global.

Na prática, a modelagem de estruturas de grande porte são realizadas em programas de simulação comerciais, com o objetivo de capturar o comportamento global da estrutura. Os efeitos localizados, quando presentes, são avaliados isoladamente de forma analítica, se possível, ou por métodos numéricos (Silveira Filho, 2023). Programas de pesquisa, como o ambiente computacional INSANE, implementam métodos numéricos mais avançados, capazes de tratar a presença de efeitos localizados de forma mais adequada e eficiente.

As estratégias de acoplamento surgem, portanto, como uma ferramenta capaz de integrar métodos numéricos avançados - presentes em programas de pesquisa - com ferramentas de simulação já existentes (programas de simulação comerciais). Como na indústria o código-fonte dos programas de simulação são fechados, ou seja, não estão disponíveis

para consulta e edição por terceiros, as estratégia de acoplamento não intrusivo, por dispensarem qualquer alteração nas aplicações envolvidas, têm maior evidência em pesquisas recentes.

Assim, o ambiente computacional INSANE, que implementa formulações e métodos numéricos avançados, mas que apresenta limitações de desempenho que impedem sua aplicação em problemas de larga escala, pode ser acoplado não intrusivamente com um *software* comercial capaz de realizar simulações numéricas com alta eficiência computacional.

Neste contexto, Silveira Filho (2023) realizou a implementação não intrusiva do MEFG^{gl} para análise de problemas multiescala ($\text{MEFG}^{gl}\text{-GLI}$) (Li *et al.*, 2021), combinando a análise do *software* comercial *Abaqus* via MEF com a análise via MEFG^{gl} no ambiente INSANE. Embora os resultados da implementação de Silveira Filho (2023) para as classes de problemas considerados tenham sido satisfatórios, o tema ainda consiste em uma linha de pesquisa recente. Sendo possível, portanto, investigar formas de melhorar a eficiência do procedimento de solução, bem como expandir a aplicação do método para outras classes de problemas, como os problemas da MFLE.

1.3 Objetivos

1.3.1 Objetivo geral

Este trabalho tem por objetivo geral contribuir com a ampliação da estratégia de acoplamento não intrusivo entre o *software* comercial *Abaqus* e o ambiente computacional INSANE através do $\text{MEFG}^{gl}\text{-GLI}$, buscando expandir a aplicação da estratégia dentro do contexto da MFLE e aprimorar a eficiência computacional.

1.3.2 Objetivos específicos

Como objetivos específicos deste trabalho pode-se citar:

- i. Implementação de uma nova abordagem da estratégia de acoplamento não intrusivo do $\text{MEFG}^{gl}\text{-GLI}$, a partir do algoritmo *staggered* proposto por Li *et al.* (2022).
- ii. Implementação de uma nova aplicação para gerenciar o acoplamento não intrusivo capaz de utilizar tanto a abordagem monolítica, proposta em Li *et al.* (2021) e implementada por Silveira Filho (2023), quanto o algoritmo *staggered*.
- iii. Adaptação da implementação que gerencia o acoplamento não intrusivo para considerar a versão do ambiente INSANE com propagação automática de trinca proposta por Fonseca (2019).
- iv. Investigação de parâmetros do MEFG^{gl} : variar o número de ciclos de iteração global-

local entre a mesoescala e a escala local e adotar diferentes tamanhos de zona *buffer* (região localizada entre o contorno do problema local e os nós enriquecidos com a solução numérica local), para avaliar a influência destes parâmetros sobre a estratégia iterativa do acoplamento não intrusivo.

- v. Entendimento e utilização de técnicas de relaxação para aceleração e garantia da convergência da estratégia iterativa do acoplamento não intrusivo.

1.4 Organização do texto

O presente trabalho está organizado em 5 capítulos e 2 apêndices, conforme descrito a seguir.

No Capítulo 1 apresenta-se uma breve introdução ao trabalho, contextualizando o tema, justificando sua realização e apresentando os objetivos gerais e específicos traçados.

O Capítulo 2 compreende uma revisão de literatura, abordando os conceitos e métodos necessários para o entendimento e desenvolvimento do trabalho. Neste capítulo é apresentado o Método dos Elementos Finitos Global-Local, a estratégia Global Local Iterativa para o Método dos Elementos Finitos (MEF-GLI), o Método dos Elementos Finitos Generalizados (MEFG), as estratégias de acoplamento para o MEFG, e finalmente, a estratégia de acoplamento não intrusivo MEFG^{gl}-GLI, que combina o algoritmo GLI com o MEFG^{gl}.

No Capítulo 3 são apresentados os aspectos computacionais, a partir da descrição geral dos procedimentos de solução da estratégia MEFG^{gl}-GLI, seguido pela descrição da implementação realizada e das novas funcionalidades adicionadas ao ambiente INSANE.

No Capítulo 4 são apresentadas as simulações numéricas realizadas com o objetivo de validar a implementação desenvolvida, bem como de permitir a investigação de parâmetros relacionados ao MEFG^{gl} sobre os procedimentos de solução da estratégia MEFG^{gl}-GLI.

No Capítulo 5 sintetiza-se as considerações finais deste trabalho, destacando às principais contribuições da pesquisa, e denotando propostas de desenvolvimento para trabalhos futuros relacionados ao tema.

Finalmente, nos Apêndices A e B são descritos, respectivamente, os requisitos e os dados de entrada da implementação proposta.

CAPÍTULO 2

REVISÃO DA LITERATURA

2.1 Método dos Elementos Finitos Global-Local

Em muitas aplicações de engenharia, a simulação de fenômenos físicos localizados, tais como soldas, trincas, e pequenas variações na geometria ou propriedade do material, é fundamental para a previsão da vida útil de serviço ou dos modos de falha de uma estrutura (Li *et al.*, 2021). Contudo, a acurácia da modelagem e da representação destes fenômenos estão intimamente vinculadas aos recursos computacionais, que limitam o uso sistêmico de modelos numéricos complexos e precisos. Um dos principais desafios em análises de elementos finitos é, portanto, fornecer soluções numéricas precisas, enquanto minimiza o custo computacional.

Felizmente, na maioria das aplicações, modelos mais simples (i.e. computacionalmente baratos) são bons o suficiente para prever o comportamento global da estrutura, enquanto que modelos específicos (i.e. computacionalmente caros) podem ser utilizados apenas em pequenas regiões de interesse, em escala local. Tal suposição permitiu o surgimento de uma grande variedade de técnicas e métodos dedicados à computação multiescala (Duval *et al.*, 2014). Essas técnicas, tais como enriquecimento de modelos de elementos finitos e modelagem multiescala, possibilitam a execução de simulações complexas que estão fora do alcance de ferramentas de análise de elementos finitos convencional, em termos de eficiência e custo computacional (Allix e Gosselet, 2020).

Uma das abordagens de modelagem multiescala¹ proposta é a análise global-local. O conceito básico desta análise é que um modelo global grosseiro pode ser usado para obter condições de contorno apropriadas para alguma região local onde uma malha mais refinada é requerida (Whitcomb, 1991). Assim, inicialmente, a solução de um problema dito “problema global”, para o qual é usada uma discretização grosseira do domínio, sem considerar possíveis fenômenos localizados, é obtida. Em seguida, define-se um “problema local” abrangendo a região onde tais fenômenos ocorrem, utilizando como condições de contorno a solução do problema global. A solução do problema local é então obtida a

¹O termo multiescala mencionado neste trabalho refere-se a diferentes escalas de discretização, e não está associado às soluções do tipo multifísico, onde as várias escalas representam o problema físico de maneira distinta.

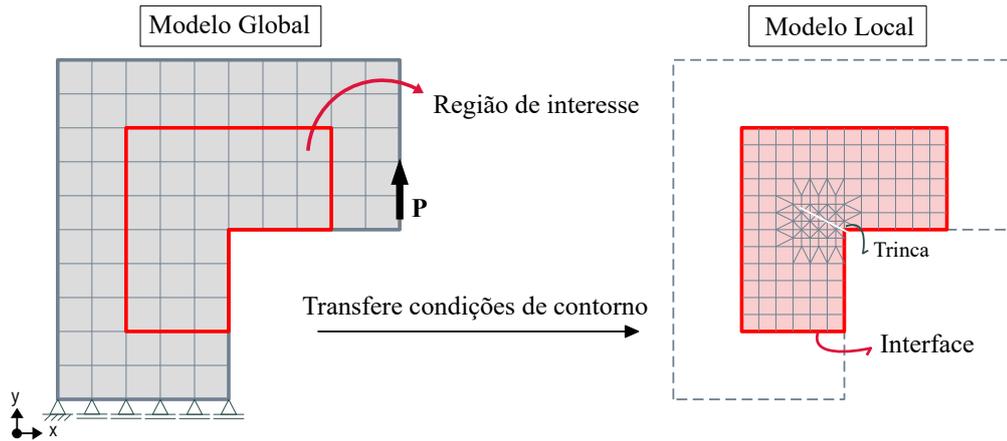


Figura 2.1: Técnica *Zooming*: Modelo global com malha grosseira é utilizado para determinar as condições de contorno a serem aplicadas ao longo da interface do modelo local refinado.

partir do refinamento da malha no domínio local, propiciando uma representação mais precisa do comportamento do fenômeno de interesse investigado.

De acordo com Noor (1986), existem quatro diferentes estratégias de análise global-local relacionadas ao MEF: técnica *zooming*; aplicação simultânea de duas técnicas de discretização; método da redução; e modelos matemáticos hierárquicos e/ou técnicas de aproximação numérica. A técnica *zooming*, também conhecida como MEF Global-Local (MEF-GL), submodelagem ou modelagem *break-out* (Li *et al.*, 2021), é a mais frequentemente utilizada no contexto de Engenharia de Estruturas (Duval *et al.*, 2014), e será, portanto, a técnica explorada aqui.

A técnica *zooming* consiste basicamente em usar uma solução global como condição de contorno para um problema local refinado. Assim, após a resolução de um problema global através de uma malha grosseira, *zooms* (termo usado no sentido figurado de uma lente para ampliação de detalhes) são aplicados em zonas críticas locais que buscam representar de maneira fidedigna os detalhes localizados. Vários níveis de refinamento podem ser estabelecidos e a compatibilidade entre a malha grosseira global e a malha refinada local pode ser obtida através de elementos de transição (Noor, 1986).

Na técnica *zooming*, portanto, são definidos dois modelos: um modelo global com uma malha grosseira, e um modelo local com uma malha refinada. Os dois modelos se “comunicam” através de interfaces que delimitam a região de interesse dentro do modelo global, de modo que, a solução do modelo global fornece as condições de contorno do modelo local (Fig. 2.1).

Em geral, os deslocamentos calculados no modelo global são utilizados como condição de contorno do modelo local. Essa abordagem fornece resultados mais precisos, sem o risco de instabilidades numéricas advindas da imposição de forças que não são necessariamente autoequilibradas. Contudo, de acordo com Curreli *et al.* (2018), condições de contorno baseadas em deslocamentos não são indicadas quando os campos de deslocamento do

modelo global e local podem diferir, isto é, quando a região de interesse possui variação de geometria e/ou em situações de problema de contato. Nestes casos, sugerem-se que as forças extraídas da solução global sejam utilizadas como condição de contorno do modelo local.

Uma das vantagens desta técnica é a de concentrar o esforço computacional apenas nas zonas que mais necessitam, além de permitir conectar programas comerciais e de pesquisa (Allix e Gosselet, 2020). No entanto, a técnica *zooming* é aplicada em apenas um sentido (do global para o local), sem considerar as repercussões dos efeitos locais no comportamento global da estrutura, podendo levar a erros de análise relevantes (Guinard *et al.*, 2018). Por exemplo, a análise de um ponto dentro da região de interesse do domínio global pode ser realizada com maior precisão a partir da correspondência deste ponto no modelo local, porém, um ponto de análise no domínio global fora da região de interesse não recebe qualquer contribuição da análise local.

Ainda, segundo Noor (1986), as principais questões e limitações relacionadas a aplicação da técnica *zooming* são: o tratamento das interfaces entre os problemas global e local; a necessidade de se transferir com razoável precisão os resultados da análise global como condições de contorno da análise local; e a incapacidade de lidar com as possíveis interações entre problemas local e global ao longo do processo de análise.

2.2 Estratégia não intrusiva MEF-GLI

O denominado Iterative Global-Local Finite Element Method (IGL-FEM), ou em português, Método dos Elementos Finitos Global-Local Iterativo (MEF-GLI), consiste em uma metodologia fortemente não intrusiva para o MEF inicialmente proposta por Whitcomb (1991). Este método pode ser interpretado como uma variação da técnica *zooming* (Li *et al.*, 2021).

O MEF-GLI adota dois modelos distintos: um modelo global discretizado por uma malha grosseira que descreve o comportamento geral da estrutura, e um modelo local que adota uma discretização refinada capturando características locais de interesse (detalhes geométricos, defeitos, trincas, inclusões, alterações no material, etc). A solução é calculada através de um processo iterativo que impõe condições de equilíbrio e continuidade entre os modelos global e local (Li *et al.*, 2021).

Assim, a ideia por trás do acoplamento não intrusivo é utilizar o modelo global para obter uma solução inicial para o domínio local e, então, utilizá-lo como ponto de partida para um processo iterativo. Na hipótese de convergência, a solução de referência final será obtida sem que seja necessária qualquer alteração na matriz de rigidez do modelo global. A solução de referência é aquela que seria obtida por um único modelo que incluísse explicitamente o refinamento local no modelo global. Tal modelo seria demasiadamente oneroso, e por isso essa solução não é desejada.

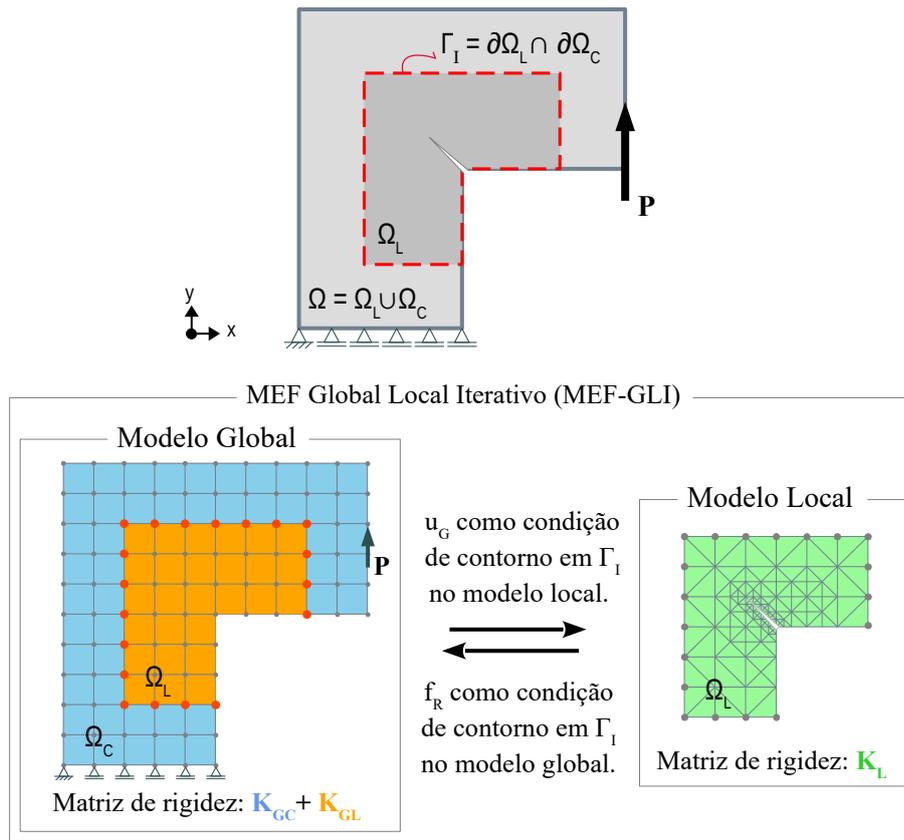


Figura 2.2: Ilustração do processo de solução do MEF Global-Local Iterativo (MEF-GLI) (adaptado de Silveira Filho (2023)).

Seja portanto, o problema apresentado na Fig. 2.2, que consiste em uma chapa em “L” com a inclusão de uma trinca na região de singularidade. O modelo global está associado ao domínio $\Omega = \Omega_L \cup \Omega_C$, em que Ω_L é o domínio local e Ω_C é o seu domínio complementar em relação ao domínio global. O modelo local, por sua vez, está restrito ao domínio Ω_L , e é definido pela região próxima à trinca, onde é esperado um comportamento mais complexo para os campos de deslocamentos e de tensões. A região $\Gamma_I = \partial\Omega_L \cap \partial\Omega_C$ é denominada de interface.

O modelo local consiste em uma discretização refinada de Ω_L , e sua rigidez é \mathbf{K}_L . Já o modelo global discretiza grosseiramente todo o domínio do problema (Ω). A rigidez do modelo global, \mathbf{K}_G , é composta por duas parcelas: a rigidez \mathbf{K}_{GL} referente à região do modelo global sobre o domínio local, e a rigidez \mathbf{K}_{GC} correspondente à região complementar.

Em análises lineares de elementos finitos a equação matricial governante pode ser expressa como:

$$\mathbf{K} \cdot \mathbf{u} = \mathbf{f} \quad (2.1)$$

A equação que governa o modelo global, aplicada ao Ω , é, portanto:

$$\mathbf{K}_G \cdot \mathbf{u}_G = \mathbf{f}_G \quad (2.2)$$

que pode ser reescrever como:

$$(\mathbf{K}_{GC} + \mathbf{K}_{GL}) \cdot \mathbf{u}_G = \mathbf{f}_G \quad (2.3)$$

Onde o sinal de “+” é a simbologia utilizada na literatura correlata, e não corresponde à soma direta entre as matrizes, mas representa a montagem final da matriz, de acordo com a posição dos graus de liberdade associados a cada matriz de rigidez.

O campo de deslocamentos \mathbf{u}_G obtido através da solução do modelo global é utilizado como condição de contorno para a solução local. Essa solução é denominada de solução global inicial, denotada por \mathbf{u}_G^0 .

A solução \mathbf{u}_L é, então, o campo de deslocamentos obtido pela resolução do modelo local, utilizando \mathbf{u}_G^0 como condição de contorno:

$$\mathbf{K}_L \cdot \mathbf{u}_L = \mathbf{f}_L \quad (2.4)$$

Os modelos global e local representam Ω_L de maneira distinta, uma vez que, em geral, $\mathbf{K}_L \neq \mathbf{K}_{GL}$. Assim, como os deslocamentos \mathbf{u}_G , calculados a partir de \mathbf{K}_{GL} , são utilizados como condição de contorno para a solução de \mathbf{u}_L , calculados a partir de \mathbf{K}_L , tem-se um desequilíbrio na interface Γ_I entre os modelos global e local (Gendre *et al.*, 2009). A partir dessa observação, Whitcomb (1991) propôs um procedimento iterativo que impõe condições de equilíbrio na interface Γ_I .

De acordo com Li *et al.* (2021), o MEF-GLI envolve os seguintes passos:

1. Análise global: resolver o modelo global, definido pela Eq. (2.3), e obter a solução inicial \mathbf{u}_G^0 ;
2. Análise local: resolver o modelo local, definido pela Eq. (2.4), considerando como condições de contorno ao longo da interface Γ_I a solução obtida do passo anterior (\mathbf{u}_G^0 na primeira iteração, e \mathbf{u}_G^i nas demais i iterações);
3. Cálculo de resíduos: utilizar as soluções \mathbf{u}_G^i e \mathbf{u}_L^i para calcular as forças internas, $\mathbf{f}_C^{\Gamma_I}$, na interface Γ_I em Ω_C e as forças de reação, $\mathbf{f}_L^{\Gamma_I}$, em Ω_L , obtendo-se um vetor resíduo, \mathbf{f}_R , definido pela diferença entre essas forças. No passo seguinte, esse vetor atualizará o vetor de forças do modelo global, portanto, será multiplicado por -1 (Silveira Filho, 2023), conforme Eq. (2.5);

$$\mathbf{f}_R = -(\mathbf{f}_L^{\Gamma_I} + \mathbf{f}_C^{\Gamma_I}) = -[(\mathbf{K}_L \cdot \mathbf{u}_L^i - \mathbf{f}_L) |_{\Gamma_I} + (\mathbf{K}_{GC} \cdot \mathbf{u}_{GC}^i - \mathbf{f}_{GC}) |_{\Gamma_I}] \quad (2.5)$$

4. Atualização do modelo global: somar o resíduo \mathbf{f}_R ao vetor de forças do modelo global na Eq. (2.3) e obter uma nova solução \mathbf{u}_G^i ;

5. Iterações global-local: retornar ao passo 2 utilizando a solução \mathbf{u}_G^i obtida no passo 4. Os passos de 2 a 4 são repetidos até que algum critério de convergência seja atingido e a solução final de referência \mathbf{u}_f seja obtida.

A solução final do MEF-GLI é, portanto, a combinação das soluções dos problemas global e local: a solução \mathbf{u}_G é considerada sobre o domínio Ω_C , e a solução \mathbf{u}_L sobre o domínio Ω_L . Em outras palavras, a solução do problema local (\mathbf{u}_L) substitui a solução do problema global (\mathbf{u}_G) sobre Ω_L , em vez de modificá-la ou enriquecê-la (Li *et al.*, 2021). Assim, desde que o MEF-GLI seja convergente, as soluções \mathbf{u}_G e \mathbf{u}_L combinadas devem representar a solução do modelo de referência, que discretiza explicitamente o modelo local no modelo global.

Como já anteriormente mencionado, o MEF-GLI é uma estratégia não intrusiva e, permite, por isso, que diferentes solucionadores sejam adotados para resolver os modelos globais e locais, sem que modificações no código-fonte sejam necessárias. O acoplamento entre os modelos se dá pela transferência de condições de contorno em termos de deslocamentos (do global para o local), e forças (do local para o global) (Silveira Filho, 2023). Tais informações são comumente utilizadas como entrada e/ou saída de dados em programas de análise via MEF (Li *et al.*, 2021).

O critério de convergência do MEF-GLI pode ser arbitrariamente definido (Silveira Filho, 2023). Whitcomb (1991), por exemplo, em seu trabalho, utilizou como critério de convergência o máximo dos módulos dos elementos do vetor resíduo \mathbf{f}_R , e indicou que tolerâncias inferiores a 10^{-7} são desnecessárias, uma vez que aumentavam o número de iterações sem aprimorar a precisão da análise. Li *et al.* (2021), Li *et al.* (2022) e Silveira Filho (2023), por sua vez, avaliaram a convergência a partir do resíduo relativo, definido pela relação entre as normas euclidianas do vetor de forças residuais na i -ésima iteração, \mathbf{f}_R^i , e o vetor de forças residuais da primeira iteração, \mathbf{f}_R^0 , e apontaram que valores de tolerância da ordem de 10^{-5} conduzem a resultados satisfatórios e suficientemente precisos.

A convergência do MEF-GLI, contudo, depende da diferença de rigidez entre \mathbf{K}_{GL} e \mathbf{K}_L . Problemas ocorrem quando os modelos global e local apresentam significativa diferença de rigidez na modelagem do domínio local. Como em geral a malha do modelo local é mais refinada do que a malha do modelo global, é comum que o problema local seja menos rígido do que o global. Neste caso, o procedimento GLI é numericamente estável (Whitcomb, 1991), e a convergência ocorre. Porém, em situações de contato ou na presença de enrijecedores, o problema local pode se tornar mais rígido do que o global, como consequência, não será obtida uma solução que satisfaça o equilíbrio de esforços na interface e, portanto, não haverá convergência.

Várias técnicas para melhorar a convergência do MEF-GLI podem ser encontrados na literatura, como por exemplo, relaxação estática e relaxação dinâmica. Essas técnicas se concentram em limitar o tamanho do passo da correção da solução global a cada iteração

(Li *et al.*, 2021), e serão melhor exploradas na Seção 2.2.1.

Outra limitação da implementação não-intrusiva do MEF-GLI, está relacionada à necessidade de aninhamento da malha local em relação a malha global: o procedimento requer que os nós da fronteira do modelo local coincidam com os nós do modelo global (Whitcomb, 1991). Na maioria das aplicações do MEF-GLI, no entanto, as malhas global e local não são compatíveis ao longo da interface, pois, em geral, a discretização local é mais refinada do que a global. O uso de multiplicadores de Lagrange e o Método de Nitsche têm sido propostos na literatura para resolver esta questão (Li *et al.*, 2021). Alternativamente, elementos de transição entre as malhas global e local também podem ser utilizados. Em todos os casos, o uso destas estratégias podem implicar no aumento da complexidade da implementação do acoplamento, e, assim, encarecer computacionalmente a estratégia (Silveira Filho, 2023).

O MEF-GLI é amplamente utilizado no contexto do MEF e, recentemente, tem sido adaptado e incorporado a outros métodos, conforme realizado por Duval *et al.* (2014), Allix e Gosselet (2020) e Li *et al.* (2021), por exemplo.

2.2.1 Convergência do GLI

A convergência do algoritmo GLI depende da diferença de rigidez entre o problema local e a região de correspondência do problema local sobre o modelo global. Quanto mais significativa é a diferença de rigidez entre os dois problemas mais lenta é a convergência (Duval *et al.*, 2014). Em muitos casos, técnicas de relaxação são utilizadas para melhorar a velocidade de convergência do algoritmo, ou até mesmo para garantir a convergência daqueles problemas onde a solução convencional não iria convergir (Silveira Filho, 2023).

De acordo com Duval *et al.* (2014), a utilização de técnicas de relaxação requer que a solução do problema global, dado pela Eq. (2.2), seja reescrita em sua forma iterativa. Assim, para uma dada iteração i , tem-se:

$$\mathbf{u}_G^{i+1} = \mathbf{u}_G^i + K_G^{-1} f(\mathbf{u}_G^i) \quad (2.6)$$

onde $f(\mathbf{u}_G^i) = -\underline{\mathbf{f}}_R^i$, e, $\underline{\mathbf{f}}_R^i$ consiste na projeção do vetor resíduo (\mathbf{f}_R), dado pela Eq. (2.5), para o espaço do problema global na iteração i .

Liu *et al.* (2014) classifica as técnicas de relaxação em dois grupos. O primeiro grupo é a família de técnicas associadas ao método quasi-Newton (Gendre *et al.*, 2009). Essas técnicas, apesar de melhorarem o comportamento da convergência, envolvem, em geral, a modificação da matriz de rigidez do problema global, o que inviabiliza sua utilização na implementação não intrusiva do algoritmo GLI.

O segundo grupo de técnicas atua somente na manipulação da função $f(\mathbf{u}_G^i)$, mantendo os demais termos da formulação do problema inalterados. Essas técnicas também melhoram o comportamento da convergência, e possuem, normalmente, implementação

mais simples, uma vez que realizam a correção da solução \mathbf{u}_G^i a partir das soluções obtidas nas iterações anteriores (Silveira Filho, 2023).

As técnicas de relaxação estática e de relaxação dinâmica fazem parte do segundo grupo de técnicas de relaxação. Essas técnicas terão suas formulações descritas nas subseções a seguir, e serão as técnicas utilizadas para melhorar o comportamento da convergência do algoritmo GLI no presente trabalho.

2.2.1.1 Relaxação estática

Entre as várias abordagens para a técnica de relaxação estática, será tratada aqui a estratégia de Xu (1992), apresentada em Silveira Filho (2023). Essa é a técnica mais simples dentre as técnicas de relaxação disponíveis.

Nesta técnica, o resultado da Eq. (2.6), passa a ser considerado como solução prevista inicialmente a ser corrigida:

$$\bar{\mathbf{u}}_G^{i+1} = \mathbf{u}_G^i + K_G^{-1} f(\mathbf{u}_G^i) \quad (2.7)$$

A relaxação é então realizada a partir da Eq. (2.8), a seguir:

$$\mathbf{u}_G^{i+1} = \mathbf{u}_G^i + \omega \Delta \mathbf{u}_G^{i+1} \quad (2.8)$$

onde $\Delta \mathbf{u}_G^{i+1} = \bar{\mathbf{u}}_G^{i+1} - \mathbf{u}_G^i$, e ω é o fator de relaxação. A Eq. (2.8) pode ser reescrita, portanto, como:

$$\mathbf{u}_G^{i+1} = \mathbf{u}_G^i + \omega(\bar{\mathbf{u}}_G^{i+1} - \mathbf{u}_G^i) \quad (2.9)$$

O fator de relaxação, ω , é um escalar predefinido e constante ao longo de todas as iterações. A predefinição de um valor ideal de ω é a principal limitação da técnica. Segundo Chevreuil *et al.* (2013), no contexto do GLI, quando a região local é mais rígida na discretização da escala local do que na escala global, valores de ω no intervalo $0 < \omega < 1,0$ garantem a convergência da solução para uma solução correta. Quando, porém, a escala global é mais rígida do que a escala local, valores de $\omega > 1,0$ aceleram a convergência da solução.

2.2.1.2 Relaxação dinâmica

A técnica de relaxação dinâmica segue as formulações da técnica de relaxação estática, mas introduz a ideia de atualizar o valor do fator de relaxação a cada iteração, isto é,

$$\mathbf{u}_G^{i+1} = \mathbf{u}_G^i + \omega^{i+1}(\bar{\mathbf{u}}_G^{i+1} - \mathbf{u}_G^i) \quad (2.10)$$

No presente trabalho, a atualização do valor de ω é realizada a partir da fórmula de

Δ^2 de Aitken, conforme proposto por Irons e Tuck (1969) e adotado em Liu *et al.* (2014), Duval *et al.* (2014), Li *et al.* (2021) e Li *et al.* (2022). Assim, definindo os vetores de incremento de deslocamentos previstos, calculados na interface Γ_I , como:

$$\Delta^{i+1} = (\bar{\mathbf{u}}_G^{i+1} - \mathbf{u}_G^i) |_{\Gamma_I}, \quad \Delta^i = (\bar{\mathbf{u}}_G^i - \mathbf{u}_G^{i-1}) |_{\Gamma_I} \quad (2.11)$$

tem-se que o fator de relaxação ω^{i+1} de uma iteração i é definido pela Eq. (2.12).

$$\omega^{i+1} = -\omega^i \frac{\Delta^i \cdot (\Delta^{i+1} - \Delta^i)}{\|\Delta^{i+1} - \Delta^i\|^2} \quad (2.12)$$

É importante observar que o cálculo de ω^{i+1} não encarece computacionalmente a solução, uma vez que a determinação de Δ^{i+1} e Δ^i utilizados no cálculo do fator de relaxação envolvem apenas os graus de liberdade na interface Γ_I , que são em pequeno número se comparadas à ordem de grandeza do problema.

De acordo com Duval *et al.* (2014), a técnica de relaxação dinâmica é mais eficiente do que a técnica de relaxação estática, mesmo quando um fator de relaxação ótimo é utilizado.

2.3 Método dos Elementos Finitos Generalizados

Para o tratamento de soluções não suaves, que envolvam descontinuidades, singularidades e elevados gradientes localizados, o já consolidado MEF apresenta limitações. A estratégia baseada na construção de espaços polinomiais para a aproximação pode exigir elevado nível de refinamento e/ou remalhamento para capturar as singularidades e os altos gradientes das grandezas envolvidas. O MEFG surge, portanto, como um método alternativo ao MEF convencional, utilizando abordagens mais adequadas para tratar essa classe de problemas.

De acordo com Duarte *et al.* (2000), a formulação do MEFG foi proposta de forma independente por Babuška e colaboradores (sob o nome de “Método dos Elementos Finitos Especiais”, “Método dos Elementos Finitos Generalizados”, e “Método da Partição da Unidade”) e por Duarte e Oden (sob o nome de “Nuvens-*hp*” e “Método das Nuvens”). O conceito fundamental da aproximação do MEFG é o enriquecimento da Partição da Unidade (PU) com funções definidas a partir do conhecimento *a priori* da solução do problema de interesse (Kim *et al.*, 2010). Assim, as funções de aproximação do MEFG são relacionadas aos pontos nodais, e construídas de modo semelhante ao Método das Nuvens-*hp*, com a diferença de que, no MEFG, a disposição dos pontos nodais é definida a partir de uma malha de elementos finitos. A utilização de funções de PU sobre uma malha de elementos finitos e o enriquecimento destas funções de modo semelhante ao Método das Nuvens-*hp*, estabelecem o MEFG como uma ponte entre o MEF convencional e os

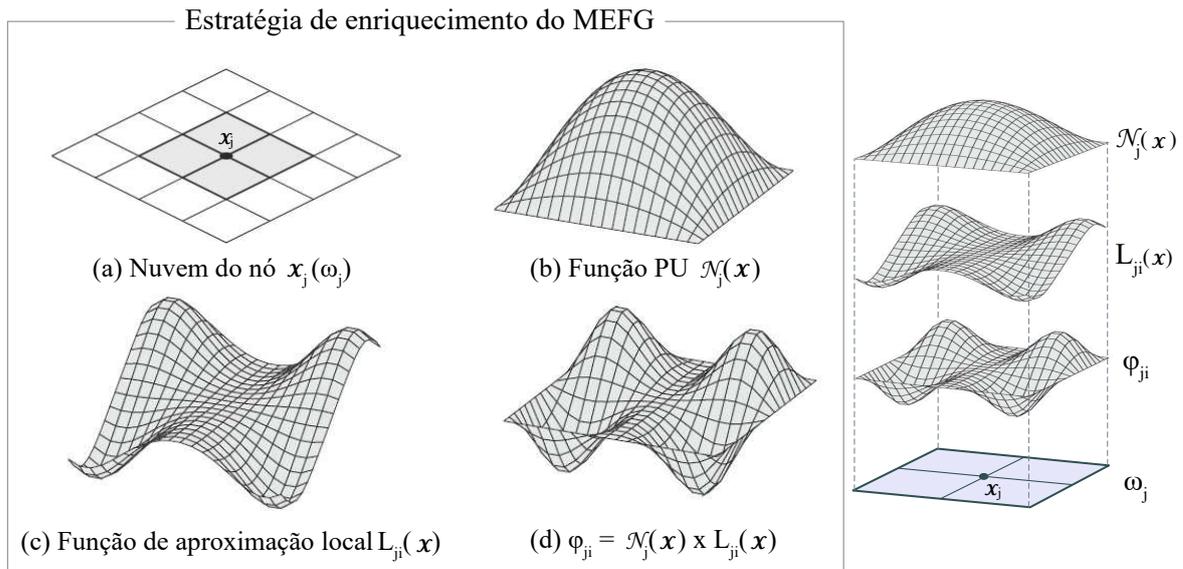


Figura 2.3: Estratégia de enriquecimento do MEFG (adaptado de Barros (2002)).

métodos sem malha (Alves, 2012).

Outro método que também emprega funções de PU do MEF convencional, e compartilha características herdadas dos métodos sem malha, é o denominado Método dos Elementos Finitos Estendido, MEFE (eXtended Finite Element Method - XFEM). Este método foi desenvolvido por Belytschko e Black (1999), em paralelo ao processo de consolidação da formulação do MEFG. Nele, apenas uma parte dos nós que constituem o modelo são enriquecidos, enquanto que no MEFG tal enriquecimento ocorria, originalmente, em todo o modelo. Com o avanço das pesquisas, as formulações do MEFG e MEFE convergiram, e são atualmente entendidos como métodos equivalentes (Fries e Belytschko, 2010), sendo referidos em muitos trabalhos simplesmente como MEFG/E (ou, em inglês, por G/XFEM).

2.3.1 Formulação do MEFG

A estratégia de construção das funções de forma do MEFG é ilustrada na Fig. 2.3, referente à aproximação de um domínio bidimensional. O termo nuvem (ω_j , na Fig. 2.3 (a)), herdado do Método das Nuvens-*hp*, representa no MEFG o conjunto de elementos finitos que compartilham o mesmo ponto nodal \mathbf{x}_j . A PU é fornecida por funções de forma do MEF convencional – a exemplo das funções Lagrangianas lineares (representadas por $\mathcal{N}_j(\mathbf{x})$ na Fig. 2.3(b)). Tais funções formam a base de uma PU, pois, para todo \mathbf{x} no domínio coberto pela malha de elementos finitos, tem-se:

$$\sum_{j=1}^n \mathcal{N}_j(\mathbf{x}) = 1 \quad (2.13)$$

De acordo com Barros (2002), a escolha das funções de forma convencionais do MEF para a PU preserva o caráter interpolador da aproximação e facilita a imposição das condições de contorno essenciais, além de evitar problemas de integração numérica comumente encontrados nos métodos sem malha.

O conjunto I_j de funções enriquecedoras (representadas na Fig. 2.3(c)) é composto por q_j funções linearmente independentes definidas para cada nó \mathbf{x}_j :

$$I_j \stackrel{def}{=} \{L_{j1}(\mathbf{x}), L_{j2}(\mathbf{x}), \dots, L_{jq_j}(\mathbf{x})\} = \{L_{ji}(\mathbf{x})\}_{i=1}^{q_j} \quad (2.14)$$

com $L_{j1}(\mathbf{x}) = 1$.

As funções $L_{ji}(\mathbf{x})$ são, a princípio, quaisquer, podendo ser polinomiais ou não, ou mesmo ser definidas a partir do conhecimento *a priori* da solução. São funções que constituem aproximações locais que devem representar bem a solução sobre o suporte a elas associado (Barros, 2002).

Finalmente, as funções de forma $\phi_{ji}(\mathbf{x})$ do MEFG (representas na Fig. 2.3(d)) são definidas pela PU enriquecida, ou seja, pelo produto entre as funções da PU e as q_j funções enriquecedoras, associadas a cada nuvem ω_j :

$$\{\phi_{ji}\}_{i=1}^{q_j} = \mathcal{N}_j(\mathbf{x}) \times \{L_{ji}(\mathbf{x})\}_{i=1}^{q_j} \quad (2.15)$$

sem somatório em j .

Utilizando as funções de forma definidas na Eq. (2.15), a aproximação genérica de uma função vetorial $\tilde{\mathbf{u}}(\mathbf{x})$ do MEFG é construída por meio da seguinte combinação linear:

$$\tilde{\mathbf{u}}(\mathbf{x}) = \sum_{j=1}^N \mathcal{N}_j(\mathbf{x}) \left\{ \mathbf{u}_j + \sum_{i=2}^{q_j} L_{ji}(\mathbf{x}) \mathbf{b}_{ji} \right\} \quad (2.16)$$

na qual \mathbf{u}_j e \mathbf{b}_{ji} são parâmetros nodais associados a cada componente $\mathcal{N}_j(\mathbf{x})$ do MEF e $\mathcal{N}_j(\mathbf{x})L_{ji}(\mathbf{x})$ do MEFG, respectivamente.

A função de forma do MEFG herda, portanto, o suporte compacto da PU e características importantes das funções enriquecedoras. Assim, ainda que a Eq. (2.16) permita que diferentes conjuntos de funções $L_{ji}(\mathbf{x})$ possam ser definidos para diferentes nós (Oden *et al.*, 1996), pela maneira como ocorre o enriquecimento, a aproximação global é construída sem penalizar a continuidade dos campos aproximadores entre os elementos, atendendo ao critério de conformidade (Barros, 2002).

2.3.2 Modelagem de Trincas através do MEFG

No tratamento de problemas da MFLE, a estratégia de enriquecimento do MEFG permite a modelagem de trincas de maneira não-geométrica (Silva, 2016), por meio de combinações de funções enriquecedoras analíticas que incorporam campos de descontinuidade

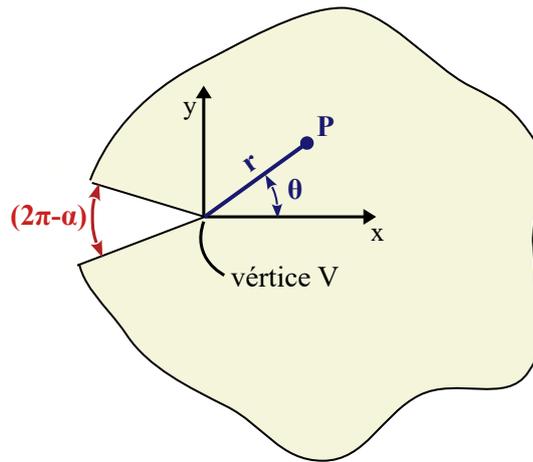


Figura 2.4: Domínio em \mathbb{R}^2 com vértice V (adaptado de Szabo e Babuška (1991)).

e singularidade à aproximação. A função de Heaviside, por exemplo, definida conforme a Eq. (2.17), corresponde a um enriquecimento do tipo degrau. Tal função introduz uma descontinuidade forte no domínio, ou seja, um salto no campo de deslocamentos (Silva, 2016):

$$H(\xi) = \begin{cases} 1 & \forall \xi > 0 \\ 0, & \forall \xi < 0 \end{cases} \quad (2.17)$$

na qual ξ representa a posição em relação à descontinuidade assumida em $\xi = 0$.

Ainda, na região da ponta da trinca, a singularidade do campo de tensões pode ser descrita através de funções que correspondem à expansão assintótica do campo de deslocamento na vizinhança de um vértice do contorno, conforme Fig. 2.4, propostas por Szabo e Babuška (1991). Estas funções são apresentadas pelas Eqs. (2.18) e (2.19) para o modo I de abertura, e nas Eqs. (2.20) e (2.21) para o modo II de abertura.

$$u_x^{(1)} = \frac{1}{2G} r^{\lambda^{(1)}} [(\kappa - Q^{(1)}(\lambda^{(1)} + 1)) \cos \lambda^{(1)} \theta - \lambda^{(1)} \cos(\lambda^{(1)} - 2)\theta] \quad (2.18)$$

$$u_y^{(1)} = \frac{1}{2G} r^{\lambda^{(1)}} [(\kappa + Q^{(1)}(\lambda^{(1)} + 1)) \sin \lambda^{(1)} \theta + \lambda^{(1)} \sin(\lambda^{(1)} - 2)\theta] \quad (2.19)$$

$$u_x^{(2)} = \frac{1}{2G} r^{\lambda^{(2)}} [(\kappa - Q^{(2)}(\lambda^{(2)} + 1)) \cos \lambda^{(2)} \theta - \lambda^{(2)} \cos(\lambda^{(2)} - 2)\theta] \quad (2.20)$$

$$u_y^{(2)} = \frac{1}{2G} r^{\lambda^{(2)}} [(\kappa + Q^{(2)}(\lambda^{(2)} + 1)) \sin \lambda^{(2)}\theta + \lambda^{(2)} \sin(\lambda^{(2)} - 2)\theta] \quad (2.21)$$

sendo:

- G o Módulo de Elasticidade Transversal;
- $\kappa = (3 - 4\nu)$ para Estado Plano de Deformações (EPD) e $\kappa = \frac{(3-\nu)}{1+\nu}$ para Estado Plano de Tensões (EPT), onde ν é o coeficiente de Poisson;
- λ e Q constantes determinadas para que a solução satisfaça ao equilíbrio e condições de contorno do problema, e dependentes do ângulo α definido na Fig. 2.4;
- r e θ coordenadas polares com origem na ponta da trinca.

Funções de enriquecimento analíticas, como as equações anteriormente mencionadas Eqs. (2.17), (2.18), (2.19), (2.20) e (2.21), são incorporadas à aproximação numérica valendo-se de um conhecimento *a priori* do comportamento físico do problema analisado, e, por isso, são viáveis apenas para uma pequena gama de problemas relevantes de engenharia. A dificuldade de prever funções enriquecedoras é especialmente verificada na análise multiescala, em problemas tridimensionais e em modelos de comportamento não-linear (Gupta, Kim e Duarte, 2012).

Ademais, de acordo com Fonseca (2019), mesmo com a introdução de funções de singularidade, a solução do MEF G para fenômenos localizados, que ocorrem, por exemplo, em meios com a presença de descontinuidade, é incapaz de extrair respostas apuradas de malhas mais grosseiras em problemas tridimensionais. Assim, a demanda pelo refinamento da malha, para controle dos erros de discretização, contrapõe-se às vantagens alcançadas pelo esquema de enriquecimento.

Este contexto leva ao desenvolvimento do Método dos Elementos Finitos Generalizados com enriquecimento global-local (MEFG^{gl}) (Duarte e Kim, 2008), que combina a estratégia global-local através do MEF, proposta por Noor (1986), com o enriquecimento da PU do MEF G . Assim, em vez de usar expressões pré-definidas como funções de enriquecimento, este método fornece uma estrutura para enriquecer o espaço de solução do problema global com funções construídas numericamente a partir da solução de um problema de valor de contorno definido localmente em uma região de interesse (Gupta, Kim e Duarte, 2012).

2.3.3 Formulação do MEF G^{gl}

Do ponto de vista da aproximação, a estratégia por trás do MEF G^{gl} baseia-se na decomposição da solução em duas escalas de análise: uma escala grosseira, capaz de representar a parcela suave da solução, e uma escala refinada, visando à descrição de características locais do problema de interesse.

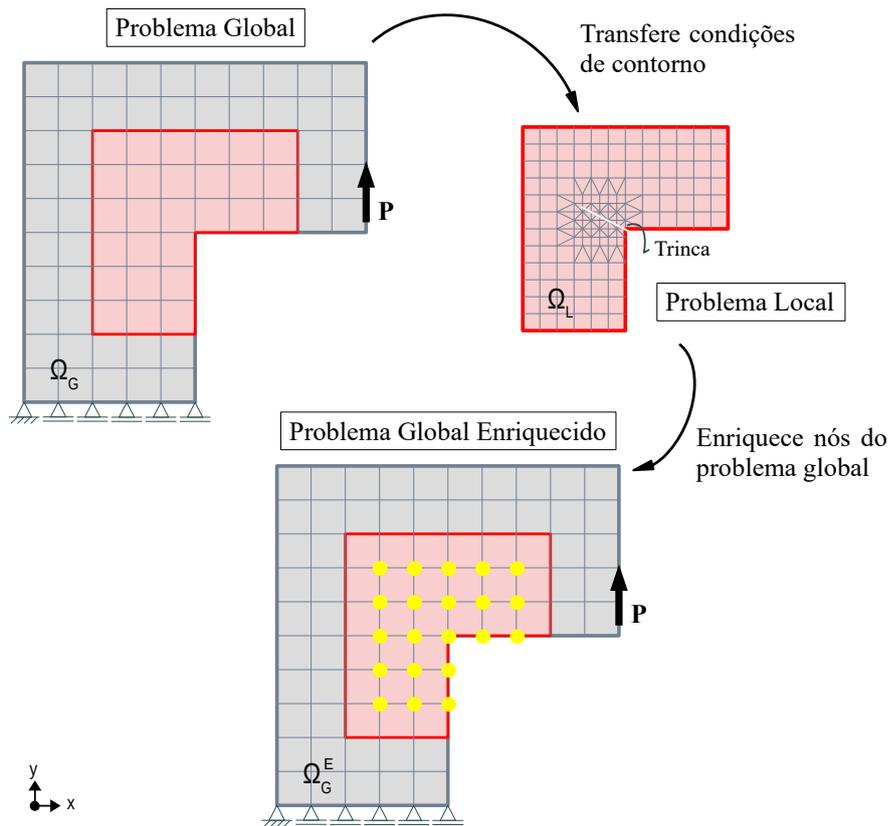


Figura 2.5: Representação esquemática do enriquecimento global-local para um problema bidimensional. Os nós globais enriquecidos estão destacados em amarelo.

A solução de um problema através do MEF G^{gl} é apresentada na Fig. 2.5, e pode ser descrita em três etapas, conforme Duarte e Kim (2008):

1. **Solução do problema global inicial:** inicialmente é obtida, via MEF convencional ou MEF G , a solução inicial do problema global ($\tilde{\mathbf{u}}_G^0$) em sua forma fraca, dada pela Eq. (2.22). Essa solução se refere ao domínio global do problema, denotado por Ω_G , e não considera os possíveis fenômenos localizados, tais como singularidades, descontinuidades, e elevados gradientes;

$$\text{Encontre } \tilde{\mathbf{u}}_G^0 \in \tilde{\mathcal{X}}_G^0(\Omega_G) \subset \mathcal{H}^1(\Omega_G) \forall \mathbf{v}_G^0 \in \tilde{\mathcal{X}}_G^0(\Omega_G)$$

$$\int_{\Omega_G} \boldsymbol{\sigma}(\tilde{\mathbf{u}}_G^0) : \boldsymbol{\varepsilon}(\mathbf{v}_G^0) d\mathbf{x} = \int_{\partial\Omega_G^\sigma} \bar{\mathbf{t}} \cdot \mathbf{v}_G^0 ds \quad (2.22)$$

onde $\boldsymbol{\sigma}$ é o tensor de tensões, $\boldsymbol{\varepsilon}$ é o tensor de deformações, $\tilde{\mathcal{X}}_G^0(\Omega_G)$ é a discretização de $\mathcal{H}^1(\Omega_G)$, um espaço de Hilbert de ordem 1, definido em Ω_G , construído com as funções de forma do método utilizado (MEF ou MEF G) e que atende às condições de contorno homogêneas de Dirichlet. O termo \mathbf{v}_G^0 refere-se à função tentativa.

2. **Solução do problema local:** um domínio local é então definido, Ω_L , contido no domínio global, isto é, $\Omega_L \subset \Omega_G$, abrangendo toda a região na qual os fenômenos

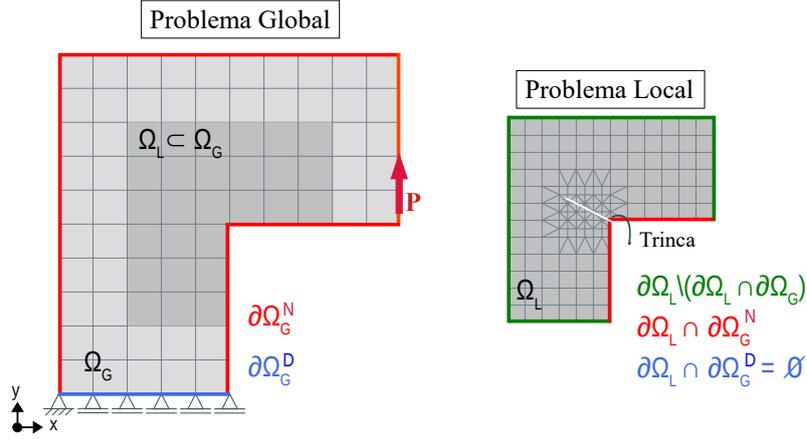


Figura 2.6: Representação esquemática do domínio global e local no MEF G^l para um problema bidimensional.

localizados de interesse ocorrem. A solução deste domínio é obtida utilizando $\tilde{\mathbf{u}}_G^0$ como condição de contorno. O problema é representado na Fig. 2.6. O contorno do domínio local ($\partial\Omega_L$) é composto por três parcelas: a parcela do contorno do domínio local que tem interseção com o contorno do domínio global e que possui as condições de contorno essenciais (condições de contorno de Dirichlet), ($\partial\Omega_L^D = \partial\Omega_L \cap \partial\Omega_G^D$); a parcela do contorno do domínio local que tem interseção com o contorno do domínio global e que possui condições de contorno naturais (condições de contorno de Neumann) ($\partial\Omega_L^N = \partial\Omega_L \cap \partial\Omega_G^N$); e, a parcela do contorno do domínio local que não possui interseção com o contorno do domínio global ($\partial\Omega_L \setminus (\partial\Omega_L \cap \partial\Omega_G)$). Assim, deve-se resolver o seguinte problema na escala local em sua forma fraca:

$$\text{Encontre } \tilde{\mathbf{u}}_L \in \tilde{\mathcal{X}}_L(\Omega_L) \subset \mathcal{H}^1(\Omega_L) \forall \mathbf{v}_L \in \tilde{\mathcal{X}}_L(\Omega_L)$$

$$\int_{\Omega_L} \boldsymbol{\sigma}(\tilde{\mathbf{u}}_L) : \boldsymbol{\varepsilon}(\mathbf{v}_L) d\mathbf{x} + \bar{\kappa} \int_{\partial\Omega_L \setminus (\partial\Omega_L \cap \partial\Omega_G)} \tilde{\mathbf{u}}_L \cdot \mathbf{v}_L ds = \int_{\partial\Omega_L \cap \partial\Omega_G^z} \bar{\mathbf{t}} \cdot \mathbf{v}_L ds + \int_{\partial\Omega_L \setminus (\partial\Omega_L \cap \partial\Omega_G)} (\mathbf{t}(\tilde{\mathbf{u}}_G^0) + \bar{\kappa} \tilde{\mathbf{u}}_G^0) \cdot \mathbf{v}_L ds \quad (2.23)$$

onde $\tilde{\mathcal{X}}_L(\Omega_L)$ é a discretização do espaço de Hilbert de ordem 1, $\mathcal{H}^1(\Omega_L)$, definido em Ω_L e utilizando as funções de forma do MEF G , $\boldsymbol{\sigma}$ é o tensor de tensões de Cauchy, $\boldsymbol{\varepsilon}$ é o tensor de pequenas deformações, \mathbf{t} é o vetor de forças generalizadas, e, $\tilde{\mathbf{u}}$ e $\bar{\mathbf{t}}$ são, respectivamente, os vetores de deslocamento e forças generalizadas prescritos. Enquanto $\bar{\kappa}$ é o parâmetro de rigidez de Cauchy, associado à imposição da solução global como condição de contorno no problema local. Assim, de acordo com Kim *et al.* (2010), é possível selecionar as condições de contorno provenientes de $\tilde{\mathbf{u}}_G^0$, que efetivamente irão atuar em Eq. 2.23, a partir da escolha do parâmetro de rigidez $\bar{\kappa}$, da seguinte forma:

- (i) $\bar{\kappa} = 0$ corresponde as condições de contorno de Neumann. São transferidas apenas tensões ao contorno $\partial\Omega_L \setminus (\partial\Omega_L \cap \partial\Omega_G)$;
- (ii) $\bar{\kappa} \gg 1$ corresponde as condições de contorno de Dirichlet, impostas pelo método da penalidade. Apenas informações de deslocamento são transferidas ao contorno $\partial\Omega_L \setminus (\partial\Omega_L \cap \partial\Omega_G)$;
- (iii) $\bar{\kappa} > 0$ corresponde as condições de contorno de Cauchy. Desse modo, é feita a transferência de tensões e deslocamentos ao contorno $\partial\Omega_L \setminus (\partial\Omega_L \cap \partial\Omega_G)$. Kim *et al.* (2010) sugerem para esses casos, com base em experimentações numéricas, a seguinte expressão para determinação de $\bar{\kappa}$:

$$\bar{\kappa} = \frac{E}{\sqrt[n]{V_0 J}} \quad (2.24)$$

sendo E o Módulo de Elasticidade Longitudinal, n o número de dimensões espaciais do problema, V_0 o volume do elemento utilizado no sistema paramétrico e J o jacobiano do elemento global cuja aresta está contida no contorno local onde são impostas as condições de contorno.

No presente trabalho a determinação de $\bar{\kappa}$ para imposição das condições de contorno de Cauchy é realizada conforme procedimento proposto em Birner e Schweitzer (2019). Tal procedimento calcula o valor de $\bar{\kappa}$ através de um balanceamento das normas de vetores calculados nos contornos global e local. Mais detalhes são descritos em Birner e Schweitzer (2019).

3. **Solução do problema global enriquecido:** finalmente, a solução $\tilde{\mathbf{u}}_L$ obtida do problema local é utilizada como enriquecimento em uma nova análise do problema global. Mantendo-se a mesma discretização do problema global inicial, procura-se representar a singularidade do problema através de $\tilde{\mathbf{u}}_L$ que enriquece a PU do problema global. Nesta etapa de análise, haverá uma i -ésima função de forma, definida por:

$$\phi_{ij} = \mathcal{N}_j(\mathbf{x}) \times \tilde{\mathbf{u}}_L \quad (2.25)$$

sendo \mathcal{N}_j a PU utilizada no problema global.

A solução do problema global-local enriquecido, $\tilde{\mathbf{u}}_G^E$, é obtida através da resolução do problema global atualizado com as funções de enriquecimento global-local, conforme Eq. (2.26).

Encontre $\tilde{\mathbf{u}}_G^E \in \tilde{\mathcal{X}}_G^E(\Omega_G) \subset \mathcal{H}^1(\Omega_G) \forall \mathbf{v}_G^E \in \tilde{\mathcal{X}}_G^E(\Omega_G)$

$$\int_{\Omega_G} \sigma(\tilde{\mathbf{u}}_G^E) : \varepsilon(\mathbf{v}_G^E) d\mathbf{x} = \int_{\partial\Omega_G^c} \bar{\mathbf{t}} \cdot \mathbf{v}_G^E d\mathbf{s} \quad (2.26)$$

As funções definadas em Eq. (2.25) são acrescentadas de maneira hierárquica ao

espaço do problema global inicial $\tilde{\mathcal{X}}_G^0(\Omega_G)$ (Li *et al.*, 2022), de modo que, o espaço $\tilde{\mathcal{X}}_G^E(\Omega_G)$ corresponde ao espaço $\tilde{\mathcal{X}}_G^0(\Omega_G)$ aumentado com as funções de enriquecimento global-local. Assim, $\tilde{\mathcal{X}}_G^E(\Omega_G)$ é dado por:

$$\tilde{\mathcal{X}}_G^E(\Omega_G) = \tilde{\mathcal{X}}_G^0(\Omega_G) \cup \left\{ \sum_{j=1} \mathcal{N}_j(\mathbf{x}) \mathbf{u}_j^{gl}(\mathbf{x}) \right\} \quad (2.27)$$

Vale destacar que $\tilde{\mathcal{X}}_G^E(\Omega_G)$ também pode ser aumentado através de outras funções de enriquecimento (Alves, 2012), além das funções de enriquecimento global-local, como, por exemplo, pelas funções enriquecedoras analíticas apresentadas na Seção 2.3.2. Ademais a característica hierárquica do método garante que, as funções de forma definidas antes do enriquecimento se mantenham inalteradas após o enriquecimento, o que favore a utilização do MEF G^{gl} em estratégias de acoplamento.

Independentemente da discretização do problema local, as funções de enriquecimento global-local adicionam a cada nó enriquecido apenas dois graus de liberdade, no caso de problemas bidimensionais, ou três graus de liberdade, no caso de problemas tridimensionais. Isso representa uma grande vantagem da estratégia do MEF G^{gl} em relação ao MEF convencional, que exige alto grau de refinamento a nível global para capturar fenômenos localizados de interesse (Kim *et al.*, 2010).

2.3.4 Ciclos global-local e Zona *buffer*

Pela forma como é definido o enriquecimento do MEF G^{gl} , a qualidade das condições de contorno aplicadas aos problemas locais influencia na precisão da solução do método (Gupta, Kim e Duarte, 2012). Como a solução do problema global inicialmente resolvido por uma discretização grosseira não é precisa, já que as características locais de interesse não são modeladas nesta escala do problema, condições de contorno de baixa precisão podem ser aplicadas ao problema local.

Gupta, Kim e Duarte (2012), ao investigarem aprimoramentos das condições de contorno oriundas do problema global para problemas tridimensionais da Mecânica da Fratura, propuseram duas estratégias para melhorar a qualidade das condições de contorno aplicadas aos problemas locais: múltiplas iterações global-local ou a zona *buffer*. O processo de múltiplas iterações global-local consiste basicamente em ciclos de iterações onde as condições de contorno de um problema local em um passo de iteração $j + 1$ são obtidas da solução do problema global enriquecido em um passo de iteração anterior, j , conforme esquematizado na Fig. 2.7.

Já a zona *buffer* corresponde a região entre a zona de enriquecimento e o limite do domínio local, ou seja, refere-se à região no problema global coincidente com o subdomínio local que não recebe enriquecimento global-local (Silveira Filho, 2023) (Fig. 2.8). Quanto

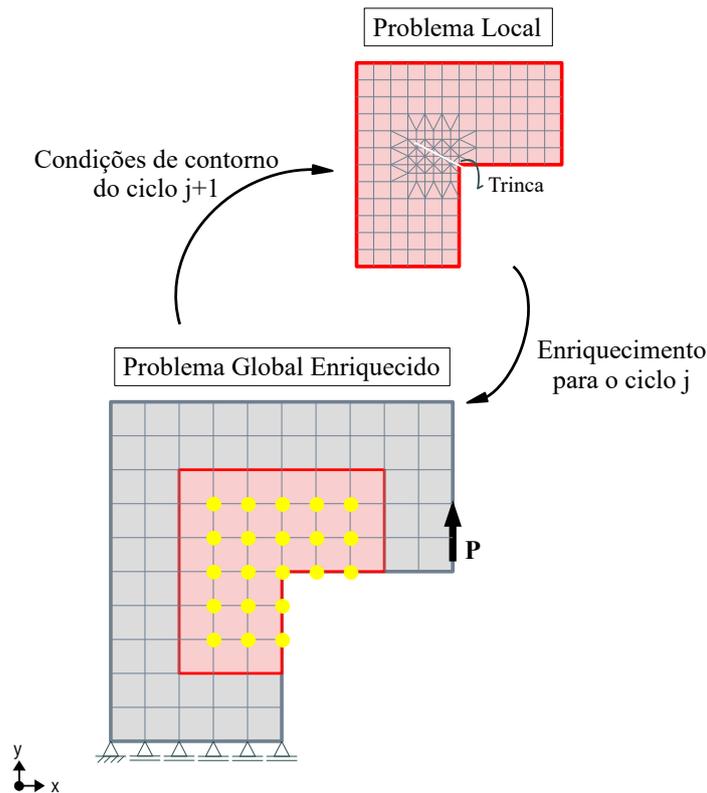


Figura 2.7: Processo de ciclo global-local no qual as condições de contorno para o problema local do ciclo $j + 1$ são obtidas da solução do problema global enriquecido do ciclo j (adaptado de Fonseca (2019)).

maior for a zona *buffer* em torno do domínio do problema local menor é o impacto de erros na construção das condições de contorno transferidas a partir do modelo global, pois ainda que dependentes, estas condições de contorno se referem aos fenômenos de interesse representados no modelo local.

De acordo com Gupta, Kim e Duarte (2012), o uso de ciclos global-local e a zona *buffer* podem levar a níveis semelhantes de erro da solução numérica local relativo ao efeito de condições de contorno de baixa precisão. Contudo, vale destacar, que, em alguns casos, a aplicação dessas estratégias podem onerar o processo de solução do problema, uma vez que podem implicar no aumento do modelo numérico da escala local, e na repetição de procedimentos de solução computacionalmente caros. Assim, cabe ao usuário escolher, com base no custo computacional e conveniência, utilizar uma ou outra, ou ambas as estratégias.

2.4 Estratégias de acoplamento para o MEFG

Simulações computacionais de programas comerciais e de pesquisa têm sido combinadas a partir de estratégias de acoplamento, com ênfase em estratégias de acoplamento não

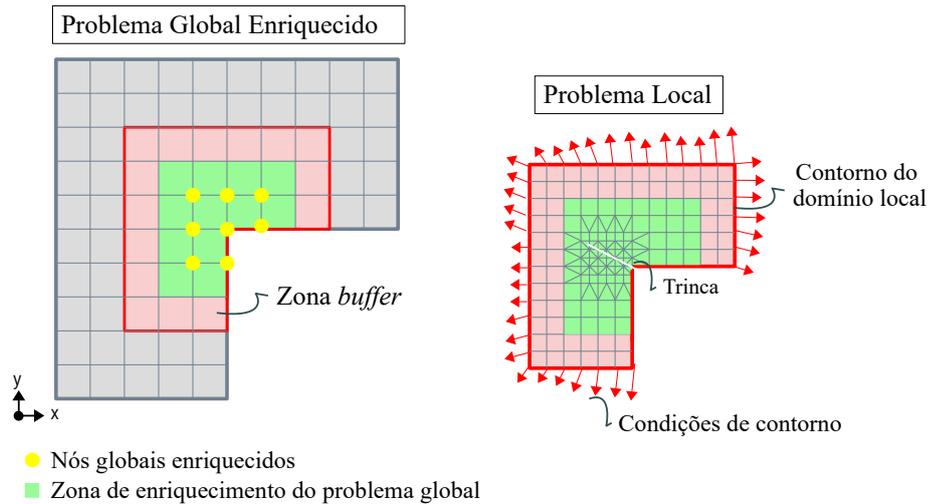


Figura 2.8: Representação esquemática da zona *buffer*.

intrusivas. Nesse contexto, nos últimos anos, algoritmos para o acoplamento não intrusivo envolvendo o MEF G , em especial o MEF G^{gl} , têm sido elaborados, com destaque para os trabalhos de Gupta, Pereira, Kim, Duarte e Eason (2012), Filmore e Duarte (2018), Li *et al.* (2021) e Li *et al.* (2022). De modo geral, as abordagens se baseiam na característica hierárquica do enriquecimento global-local, mencionada na Seção 2.3.3.

Segundo Duarte e Kim (2008), a característica hierárquica do MEF G^{gl} permite com que, as funções de forma do MEF convencional, definidas antes do enriquecimento, se mantenham inalteradas após o enriquecimento. Torna-se possível, portanto, dissociar a contribuição da solução oriunda do MEF daquela proveniente do enriquecimento, e, assim, realizar o acoplamento entre a solução inicial e a solução via MEF G^{gl} , de forma não intrusiva (Silveira Filho, 2023).

Gupta, Pereira, Kim, Duarte e Eason (2012) apresentaram um método de acoplamento não intrusivo para análise de problemas da Mecânica da Fratura no espaço tridimensional. Neste trabalho, o problema global é resolvido por um *software* comercial via MEF, os problemas locais são resolvidos por um *software* que implementa o MEF G , e o problema global enriquecido é resolvido pela combinação dos *solvers* do MEF e do MEF G . Através de uma estratégia semelhante, Plews *et al.* (2012) solucionaram problemas de gradientes térmicos elevados. Em ambos os casos, o acoplamento entre os modelos é feito por meio da condensação estática do sistema de equações do problema global.

Nessas abordagens, as interações entre os *solvers* são limitadas à troca de vetores de força e deslocamentos nodais, sem a necessidade de se realizar qualquer alteração no código-fonte das aplicações, caracterizando-se como um acoplamento não intrusivo. Contudo, para se realizar a condensação estática do sistema de equações, a menos que se tenha acesso a matriz de rigidez do problema global, ou à sua inversa, faz-se necessário aplicar pseudo forças oriundas da solução local ao problema global, tornando o procedimento oneroso. A determinação do vetor de forças requer o manuseio de vetores com dimensão

da ordem de grandeza do problema global, podendo gerar, por isso, consumo de memória elevado, e inviabilizar a simulação do comportamento de grandes estruturas, com elevado número de graus de liberdade. Tais constatações limitam, portanto, a disseminação dessa metodologia no meio industrial.

Filmore e Duarte (2018) apresentaram, posteriormente, uma alternativa de algoritmo de acoplamento não intrusivo que dispensa a condensação estática do sistema de equações do problema global, sem comprometer sua precisão e flexibilidade. Esse algoritmo, denominado *Hierarchical Non-Intrusive Algorithm* (HNA), acopla não intrusivamente um *software* de análise via MEF, comercial ou não, com um código de pesquisa que implementa um enriquecimento hierárquico de espaços de elementos finitos. Assim, o único requisito é que o *software* MEF seja capaz de exportar a matriz de rigidez e o vetor de forças do problema global, para que possam ser importados e lidos pelo *software* de análise via MEF. A implementação do HNA é mais simples do que as implementações propostas por Gupta, Pereira, Kim, Duarte e Eason (2012) e Plews *et al.* (2012), além de reduzir a demanda computacional, porém, essa metodologia requer que o tipo de elemento finito a ser utilizado esteja disponível na biblioteca de elementos dos dois programas envolvidos. Ademais, persiste a necessidade de acesso à matriz rigidez do modelo global, que nem sempre está disponível.

Finalmente, Li *et al.* (2021) propuseram uma estrutura computacional multiescala, que combina o MEF-GLI, descrito na Seção 2.2, com o MEF G^{gl} , formulado na Seção 2.3.3, denominada MEF G^{gl} -GLI. Tal estrutura, que será melhor apresentada na Seção 2.5, permite o acoplamento não intrusivo de diferentes *solvers* para realizar simulações multiescala. A estratégia não apresenta limitações de memória e nem mesmo de biblioteca de elementos, como mencionado nas estratégias anteriores. Exige-se apenas a transferência dos vetores de deslocamentos e forças generalizadas. Sua precisão é similar à obtida através de uma análise direta via MEF, e sua implementação é relativamente simples, ainda que permita a associação entre programas comerciais e de pesquisa. A maior limitação desse método está, *a priori*, associada a sua convergência lenta, ou não convergência, em alguns casos (Silveira Filho, 2023).

Mais recentemente, Li *et al.* (2022) utilizaram a estratégia proposta em Li *et al.* (2021) para acoplar modelos sólidos e de cascas tridimensionais. Baseado-se no MEF G^{gl} -GLI, propuseram o algoritmo *staggered*, para a resolução de problemas acoplados definidos nas escalas global, meso e local. O algoritmo proposto requer significativamente menos simulações MEF G^{gl} , do que o algoritmo apresentado em Li *et al.* (2021), o que o torna mais eficiente computacionalmente. Este algoritmo também será melhor descrito na Seção 2.5, a seguir.

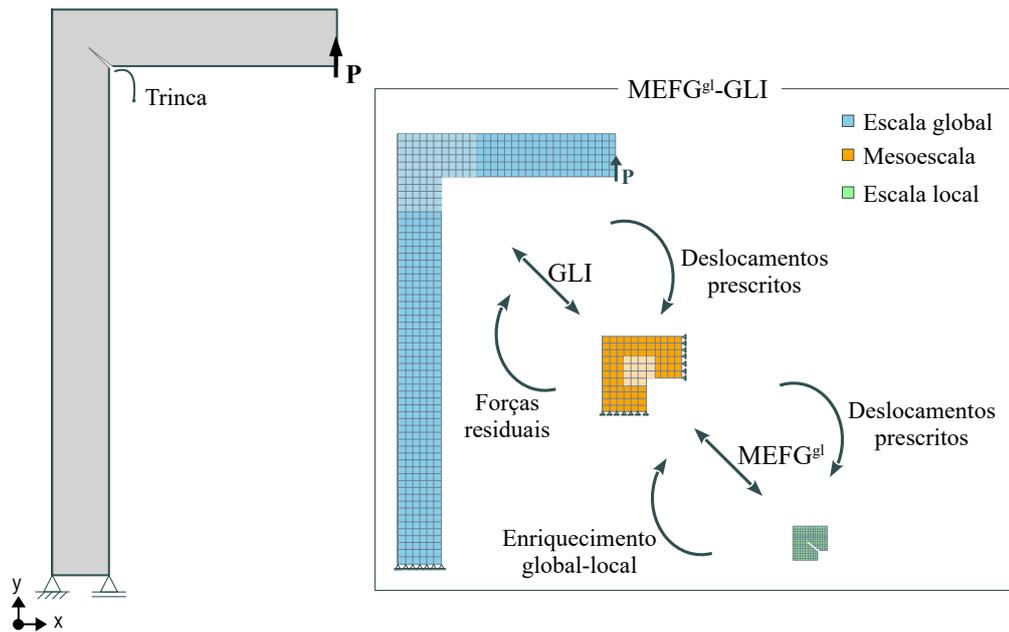


Figura 2.9: Representação esquemática da abordagem da estratégia $MEFG^{gl}$ -GLI em um pórtico em “L” com a inclusão de uma trinca.

2.5 Estratégia não intrusiva $MEFG^{gl}$ -GLI

A estratégia $MEFG^{gl}$ -GLI, inicialmente proposta por Li *et al.* (2021), consiste em uma estrutura computacional multiescala, não intrusiva, que combina o algoritmo iterativo global-local do MEF-GLI com o $MEFG^{gl}$. Uma esquematização da estratégia é apresentada na Fig. 2.9. Assim, no $MEFG^{gl}$ -GLI, o algoritmo GLI acopla um modelo global com um modelo da mesoescala, que representa um subdomínio de interesse, enquanto que o $MEFG^{gl}$ realiza o enriquecimento global-local da mesoescala a partir da solução de um modelo local. Desse modo, o problema local na análise GLI corresponde ao problema global da análise via $MEFG^{gl}$. A combinação desses métodos divide, portanto, o problema em três escalas e seus respectivos modelos e discretizações: um modelo global, um modelo da mesoescala e um, ou vários, modelos locais.

O modelo global, que não representa os fenômenos localizados de interesse, é gerado para capturar o comportamento global em escala grosseira do problema. O modelo local, por sua vez, que utiliza malha mais refinada, é criado para representar os fenômenos locais de interesse em regiões críticas. Já a mesoescala serve como um modelo de transição entre os modelos global e local, e representa a subregião do problema global que envolve o(s) fenômeno(s) localizado(s).

Em geral, a malha que discretiza a mesoescala é a mesma que discretiza o modelo global, o que facilita a transferência de informações entre os dois modelos e elimina a limitação da implementação não-intrusiva do GLI (de aninhamento entre as malhas global e local), sem a necessidade de introduzir técnicas para compatibilizar a malha na interface

entre os dois modelos. Destaca-se ainda, que, embora a mesoescala adote uma malha grosseira, certos detalhes localizados também podem ser representados nessa região, a depender da escala de comprimento desses detalhes (Li *et al.*, 2021).

De acordo com Li *et al.* (2021) e Silveira Filho (2023), os principais passos da estratégia MEFG^{gl}-GLI são, portanto:

1. Análise do modelo global: resolver o modelo global e obter a solução \mathbf{u}_G^i . Na primeira iteração, para $i = 0$, a solução global inicial é denotada por \mathbf{u}_G^0 ;
2. Análise do modelo da mesoescala: resolver o modelo da mesoescala e obter a solução \mathbf{u}_M^i utilizando a solução \mathbf{u}_G^i como condição de contorno, Eq. (2.22);
3. Análise do(s) modelo(s) local(is): resolver os diversos modelos locais existentes utilizando a solução \mathbf{u}_M^i como condição de contorno, Eq. (2.23). A solução \mathbf{u}_L^i obtida é utilizada na construção de funções de enriquecimento global-local da mesoescala;
4. Análise do modelo enriquecido da mesoescala: resolver novamente o modelo da mesoescala, que agora possui nós enriquecidos por funções construídas a partir da solução \mathbf{u}_L^i , e obter a solução \mathbf{u}_{MEnr}^i , Eq. (2.26). Ciclos global-local do MEFG^{gl}, ou seja, passos 3 e 4, podem ocorrer nesta etapa para tratar os efeitos de condições de contorno de baixa precisão;
5. Cálculo do vetor resíduo: obter, a partir das soluções \mathbf{u}_{MEnr}^i e \mathbf{u}_G^i , as reações na(s) interface(s) entre o modelo enriquecido da mesoescala e o modelo global e calcular o vetor resíduo conforme Eq. (2.5);
6. Verificação da convergência: se houver convergência da solução, \mathbf{u}_G^i e \mathbf{u}_{MEnr}^i são as soluções finais. Se não houver convergência, o vetor de forças do modelo global é atualizado pelo vetor resíduo;
7. Início de uma nova iteração: retornar ao passo 1, dando início a uma nova iteração $i = i + 1$. Os passos se repetem até o critério de convergência ser atingido.

Assim, na estratégia MEFG^{gl}-GLI, o GLI é aplicado entre os passos 1 e 2, e entre os passos 5 e 6, enquanto a análise via MEFG^{gl} é aplicada do passo 3 ao passo 4. Essa abordagem é nomeada em Li *et al.* (2022) por abordagem de solução monolítica, e tem seus principais passos esquematizados no fluxograma da Fig. 2.10.

Baseando-se na estratégia MEFG^{gl}-GLI, Li *et al.* (2022) propuseram um novo algoritmo de solução, denominado algoritmo *staggered*, que também combina o GLI com o MEFG^{gl}. A diferença é que, neste caso, os enriquecimentos global-local fornecidos pela solução do problema local são atualizados apenas após as iterações GLI entre o modelo global e a mesoescala terem atingido algum critério de convergência. Assim, o algoritmo proposto por Li *et al.* (2022) requer menos simulações MEFG^{gl} do que a estratégia proposta por Li *et al.* (2021), sendo, portanto, computacionalmente mais eficiente.

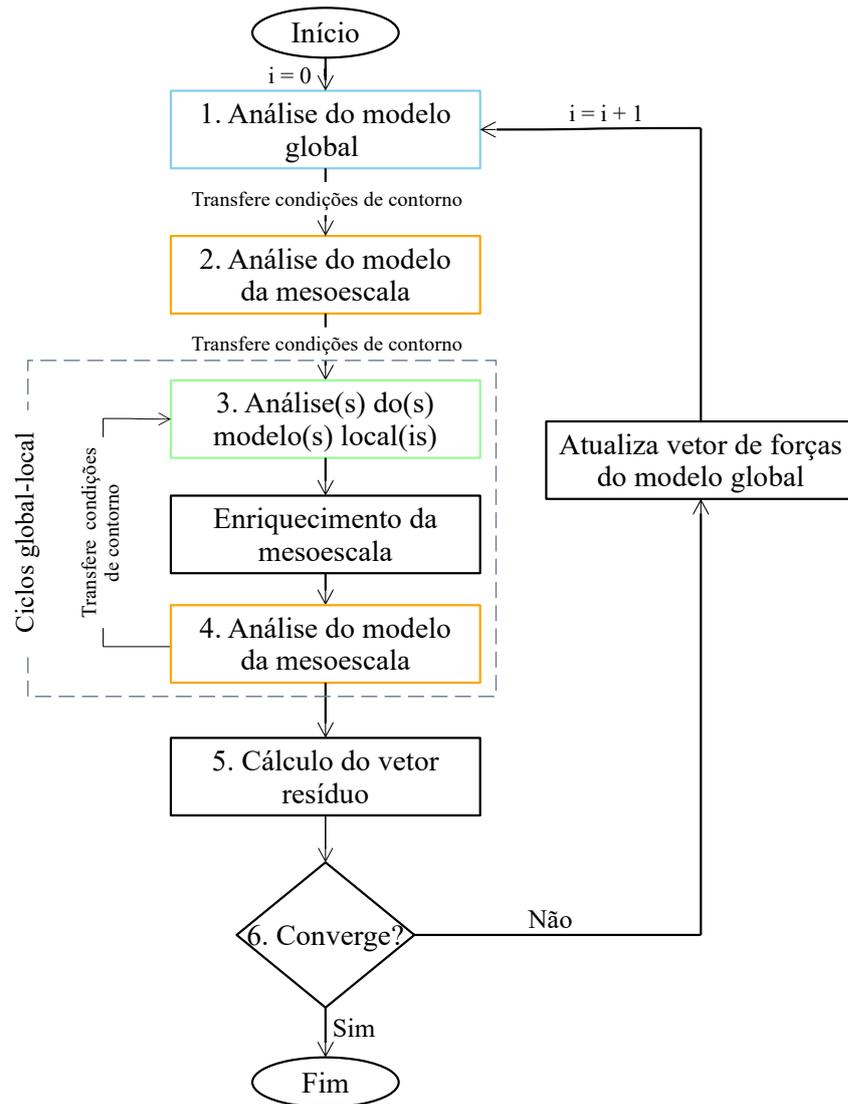


Figura 2.10: Fluxograma dos principais passos da estratégia $MEFG^{gl}$ -GLI segundo abordagem monolítica.

Os principais passos do algoritmo de solução *staggered* da estratégia MEFG^{gl}-GLI são esquematizados no fluxograma da Fig. 2.11, e podem ser descritos, conforme Li *et al.* (2022), como:

1. Análise do modelo global: resolver o modelo global e obter a solução \mathbf{u}_G ;
2. Análise do modelo da mesoescala: utilizando a solução \mathbf{u}_G , resolver o modelo da mesoescala e obter a solução \mathbf{u}_M^0 , quando $i = 0$, Eq. (2.22), ou \mathbf{u}_M^E , quando $i \neq 0$, Eq. (2.26);
3. Cálculo do vetor resíduo: obter, a partir das soluções \mathbf{u}_M^0 (se $i = 0$) ou \mathbf{u}_M^E (se $i \neq 0$), e \mathbf{u}_G , as reações na(s) interface(s) entre o modelo da mesoescala e o modelo global e calcular o vetor resíduo conforme Eq. (2.5);
4. Verificação da convergência: se houver convergência o procedimento passa para o passo seguinte. Se não houver convergência, o vetor de forças do modelo global é atualizado pelo vetor resíduo, uma nova iteração GLI é iniciada, $i_{gli} = i_{gli} + 1$, o procedimento retorna ao passo 1 e se repete até que o critério de convergência seja atingido;
5. Análise do(s) modelo(s) local(is): resolver os diversos modelos locais existentes utilizando a solução \mathbf{u}_M^0 ou a solução \mathbf{u}_M^E , convergidas do passo 4, como condição de contorno, Eq. (2.23). A solução \mathbf{u}_L obtida é utilizada na construção de funções de enriquecimento global-local da mesoescala;
6. Análise do modelo enriquecido da mesoescala: resolver novamente o modelo da mesoescala e obter a solução \mathbf{u}_M^E , Eq. (2.26). Ciclos global-local do MEFG^{gl}, ou seja, os passos 3 e 4, podem ocorrer nesta etapa para tratar os efeitos de condições de contorno de baixa precisão;
7. Cálculo do vetor resíduo: obter, a partir das soluções \mathbf{u}_M^E e \mathbf{u}_G , as reações na(s) interface(s) entre o modelo enriquecido da mesoescala e o modelo global e calcular o vetor resíduo conforme Eq. (2.5);
8. Verificação da convergência: se houver convergência da solução, \mathbf{u}_G e \mathbf{u}_M^E são as soluções finais. Se não houver convergência, o vetor de forças do modelo global é atualizado pelo vetor resíduo;
9. Início de uma nova iteração: retornar ao passo 1, dando início a uma nova iteração $i = i + 1$. Os passos se repetem até o critério de convergência ser atingido.

Li *et al.* (2022), ao comparar as soluções obtidas pelas duas abordagens para a resolução de um mesmo exemplo numérico (junta T soldada com trinca), observaram que ambas as estratégias convergem, e apresentam níveis de precisão quase idênticos em termos de deslocamento e fatores de intensidade de tensão. Contudo, embora não se tenha verificado uma diferença significativa de número de iterações GLI necessárias para convergência de cada estratégia, foi observado que a solução obtida pelo algoritmo *staggered*

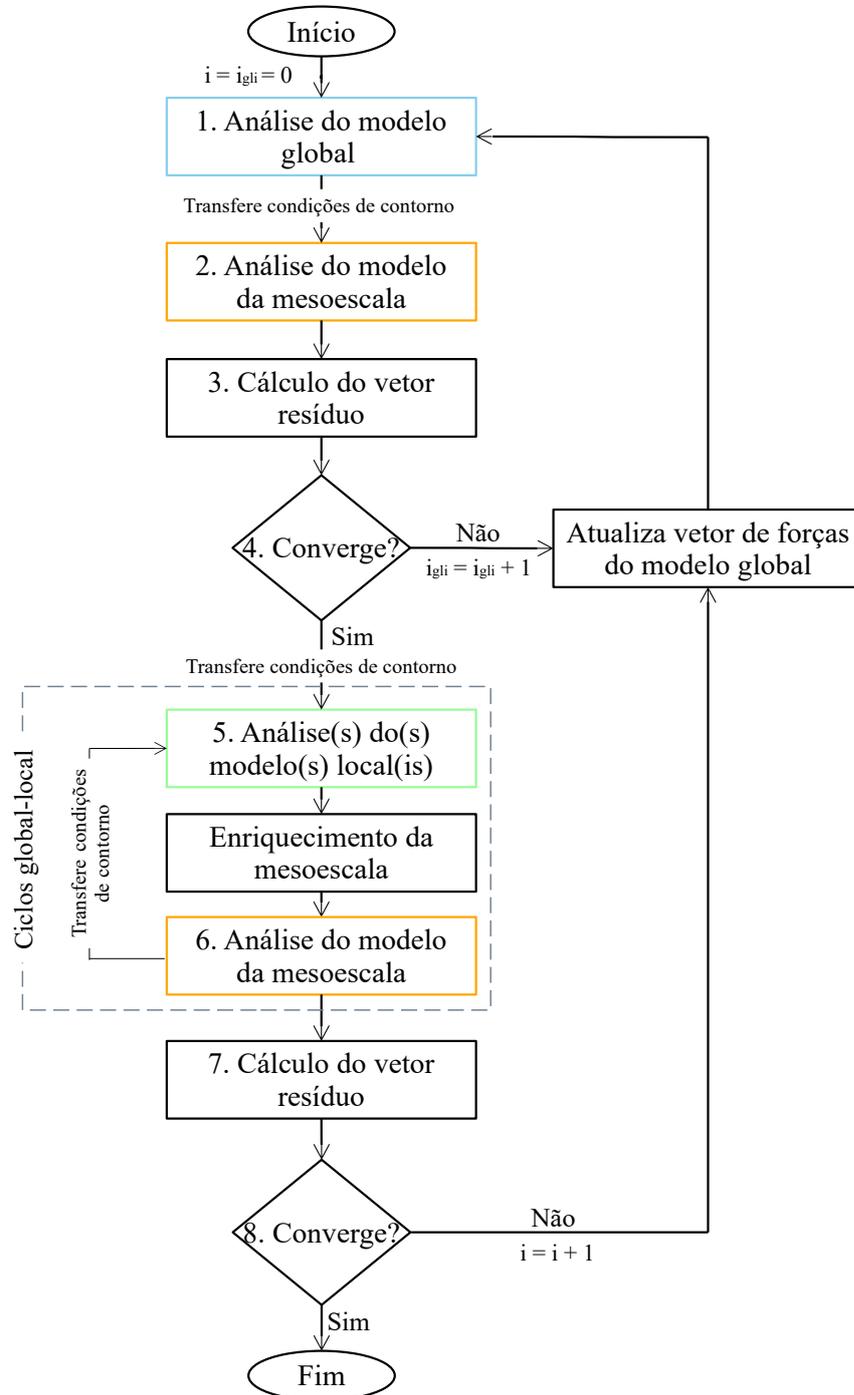


Figura 2.11: Fluxograma dos principais passos da estratégia MEFGL-GLI segundo abordagem do algoritmo *staggered*.

apresentou uma redução de 75% de tempo computacional total em relação a solução obtida em Li *et al.* (2021). Esta redução de tempo computacional é, principalmente, atribuída ao menor número de simulações MEFG^{gl} requeridas pelo algoritmo *staggered*.

CAPÍTULO 3

ASPECTOS COMPUTACIONAIS

Neste capítulo, são apresentados os principais aspectos da implementação computacional desenvolvida para o acoplamento não intrusivo do *software* comercial *Abaqus* com o ambiente computacional INSANE. Os algoritmos das abordagens monolítica e *staggered* da estratégia de acoplamento MEFG^{gl} -GLI são apresentados, seguido pela justificativa e descrição geral da implementação dos algoritmos no ambiente INSANE, pontuando as modificações realizadas na base de dados computacionais já existente e as novas funcionalidades adicionadas ao sistema.

3.1 Procedimentos de solução do MEFG^{gl} -GLI

Conforme apresentado anteriormente, o MEFG^{gl} -GLI é uma estratégia de acoplamento que combina o algoritmo GLI com o MEFG^{gl} , dividindo a análise em três escalas e seus respectivos modelos. De modo que, o acoplamento entre as escalas ocorre de forma não intrusiva, através da troca de informações em termos de deslocamento e força residual.

Na implementação proposta no presente trabalho, o modelo da escala global é solucionado pelo *software* comercial *Abaqus* via MEF, enquanto que o enriquecimento do modelo da mesoescala a partir da solução do modelo da escala local é realizado pelo MEFG^{gl} implementado no ambiente INSANE. Através da propriedade não intrusiva do GLI o algoritmo de acoplamento, juntamente com as técnicas de relaxação, foram implementados no INSANE para acoplar o modelo da escala global com o modelo da mesoescala.

Assim, diferentemente de Silveira Filho (2023), que desenvolveu uma aplicação independente para acoplar o *software* *Abaqus* com o ambiente INSANE, aqui, a não intrusividade da implementação é somente em relação ao *software* comercial. O desenvolvimento do algoritmo que gerencia o acoplamento dentro do próprio ambiente INSANE permitiu a implementação de uma abordagem computacionalmente mais eficiente da estratégia.

Serão apresentados a seguir o algoritmo de solução monolítica e o algoritmo de solução *staggered* para a estratégia não intrusiva MEFG^{gl} -GLI. Ambas abordagens de solução foram implementadas neste trabalho.

3.1.1 Algoritmo de solução monolítica

A abordagem de solução monolítica para a estratégia MEF gl -GLI foi inicialmente proposta por Li *et al.* (2021), e pode ser implementada conforme Algoritmo 3.1.

Algoritmo 3.1: Solução monolítica para o MEF gl -GLI.

Entrada:

- Modelos numéricos referentes as escalas global, meso e local;
- Lista de nós da interface entre os modelos global e meso;
- Lista de elementos da interface entre os modelos global e meso;
- Tipo de técnica de relaxação e coeficiente de relaxação (ω);
- Número de ciclos global-local (n_{GL});
- Tolerância do GLI (ϵ).

Saída:

- Solução final do modelo da escala global;
- Solução final do modelo da escala meso.

```

1   $i \leftarrow 0$ ;
2  Calcular a solução inicial do modelo global :  $\mathbf{u}_G^0$ ;
3  enquanto não convergir faça
4      se  $i > 0$  então
5           $\bar{\mathbf{u}}_G^i = \mathbf{u}_G^{i-1} + \mathbf{K}_G^{-1} \mathbf{f}_R^i$  ; /* solução prevista */
6          se relaxação dinâmica então
7              Calcular  $\omega^i$  pela Eq. (2.12) ; /* atualização de  $\omega$  */
8          senão
9               $\omega^i = \omega$  ;
10         fim
11          $\mathbf{u}_G^i = \mathbf{u}_G^{i-1} + \omega^i (\bar{\mathbf{u}}_G^i - \mathbf{u}_G^{i-1})$  ; /* solução corrigida */
12     senão
13          $\mathbf{u}_G^i = \bar{\mathbf{u}}_G^i$  ; /* não há correção na primeira iteração */
14     fim
15     Impor  $\mathbf{u}_G^i$  como condição de contorno em  $\Gamma_I$  na mesoescala;
16     Calcular a solução do modelo da mesoescala,  $\mathbf{u}_M^i$ . Se  $i = 0$  ou abordagem MEF $^{gl}$ -GLI $^{M1}$ 
        solução da mesoescala inicial. Se  $i > 0$  e abordagem MEF $^{gl}$ -GLI $^{M2}$  solução da
        mesoescala enriquecida;
17     para  $j = 0$  até  $n_{GL}$  faça
18         Impor  $\mathbf{u}_M^i$  (se  $j = 0$ ) ou  $\mathbf{u}_{MEnr}^i$  (se  $j \neq 0$ ) como condição de contorno do modelo local;
19         Calcular a solução do modelo local:  $\mathbf{u}_{L,j}^i$ ;
20         Enriquecer o modelo da mesoescala a partir da solução  $\mathbf{u}_{L,j}^i$ ;
21         Calcular a solução do modelo enriquecido da mesoescala:  $\mathbf{u}_{MEnr,j}^i$ ;
22     fim
23     Calcular  $\mathbf{f}_R^i$  pela Eq. (2.5) utilizando  $\mathbf{u}_G^i$  e  $\mathbf{u}_{MEnr,n_{GL}}^i$ ;
24     se há convergência então
25         interromper;
26     fim
27      $i \leftarrow i + 1$ ;
28 fim

```

Os modelos computacionais são, portanto, dados de entrada do procedimento de solução da estratégia, e devem ser compatíveis com os seus respectivos *solvers*. Os elementos e os nós da interface entre o modelo global e o modelo da mesoescala também devem ser identificados durante a entrada de dados, uma vez que é a partir dos nós da interface que ocorre a transferência de informação entre as duas escalas. Assim, devido ao aninhamento das malhas, há um mapeamento de um-para-um entre os nós do modelo global e de mesoescala ao longo da interface.

A técnica de relaxação também deve ser indicada. Ela é aplicada na correção da solução prevista do modelo global (Linha 11) para melhorar e/ou acelerar a convergência do GLI, e pode ser do tipo estático ou dinâmico. Na relaxação estática, após a primeira iteração, o fator de relaxação (ω) de entrada é utilizado ao longo de todas as iterações. Já na relaxação dinâmica, a atualização do valor de ω (Linha 7) requer as soluções de duas iterações anteriores ao passo corrente, logo, é necessário arbitrar os valores iniciais do procedimento, fazendo $\mathbf{u}_G^0 = \bar{\mathbf{u}}_G^0$ e $\mathbf{u}_G^1 = \bar{\mathbf{u}}_G^1$, isto é, nas duas primeiras iterações $\omega^0 = \omega^1 = 1, 0$.

As abordagens MEFG^{gl}-GLI^{M1} e MEFG^{gl}-GLI^{M2} da solução monolítica da estratégia MEFG^{gl}-GLI (Linha 16), se referem, respectivamente, ao algoritmo proposto por Li *et al.* (2021) e implementado por Silveira Filho (2023), e ao algoritmo proposto no presente trabalho. Mais detalhes serão dados na Seção 4.

O número de ciclos global-local (n_{GL}) atua na solução do MEFG^{gl} entre os modelos da escala meso e local (Linhas 17 a 22), e também deve ser informado pelo usuário. Nos exemplos apresentados em Li *et al.* (2021), $n_{GL} = 1$ foi suficiente para tratar problemas de condições de contorno de baixa precisão. Contudo, é importante notar que as iterações MEFG^{gl} podem alterar a rigidez do modelo da mesoescala, e, dessa forma, afetar a taxa de convergência do GLI. Detalhes a este respeito também serão discutidos na Seção 4, com base nos experimentos numéricos.

A tolerância do GLI (ϵ) é o dado de entrada relacionado ao critério de convergência da estratégia. Neste trabalho a convergência é verificada à partir de um resíduo relativo, dado pela Eq. (3.1).

$$\frac{\|f_R^i\|}{\|f_M^0\|} < \epsilon \quad (3.1)$$

onde $\|f_M^0\|$ é a norma euclidiana do vetor de reações do modelo da mesoescala na iteração inicial ($i = 0$), enquanto $\|f_R^i\|$ é a norma euclidiana do vetor resíduo (f_R) na iteração i corrente. A convergência ocorre quando o resíduo relativo é inferior ao valor predefinido de tolerância (ϵ).

Assim, após leitura dos dados de entrada, o algoritmo consiste basicamente em repetir os passos descritos na Seção 2.5 até o critério de convergência ser atingido. Ainda, é importante observar que, em simulações com propagação de trinca ou carregamentos

cíclicos, o Algoritmo 3.1 é realizado para cada etapa de carregamento ou para cada passo de propagação de trinca. Dessa forma, a estratégia pode exigir diferentes números de iterações (i) a cada etapa de análise.

3.1.2 Algoritmo de solução *staggered*

A abordagem de solução *staggered* para a estratégia MEFG^{gl}-GLI é um desdobramento da abordagem monolítica, e foi proposta em Li *et al.* (2022). Sua implementação pode ser realizada conforme Algoritmo 3.2.

Nessa abordagem os dados de entrada são os mesmos daqueles da abordagem monolítica, e por isso, valem as mesmas observações anteriormente descritas. A maior diferença entre as duas abordagens de solução é que, no Algoritmo 3.2 a solução do modelo da escala local, e o consequente enriquecimento global-local do modelo da mesoescala só ocorre após a convergência do GLI entre os modelos da escala global e meso terem atingido o critério de convergência. Diferentemente do que ocorre no Algoritmo 3.1, onde, o problema local e a solução MEFG^{gl} são atualizados a cada iteração.

Ademais, como na abordagem de solução *staggered* a verificação da convergência ocorre após as iterações GLI e ao final das simulações MEFG^{gl}, o Algoritmo 3.2 possui dois contadores: contador de iterações GLI (i_{gli}) e contador de iterações *staggered* (i_s). Dessa forma, a verificação da convergência para as iterações *staggered* e GLI são dadas, respectivamente, por:

$$\frac{\|f_R^{i_s}\|}{\|f_M^0\|} < \epsilon_s, \quad \frac{\|f_R^{i_{gli}}\|}{\|f_M^0\|} < \epsilon_{gli} \quad (3.2)$$

onde $\|f_M^0\|$ é a norma euclidiana do vetor de reações do modelo da mesoescala na iteração inicial, enquanto $\|f_R^{i_s}\|$ e $\|f_R^{i_{gli}}\|$ são as normas euclidiana do vetor resíduo (f_R) nas iterações i_s e i_{gli} correntes. ϵ_s e ϵ_{gli} são valores predefinidos de tolerância, e no presente trabalho $\epsilon_s = \epsilon_{gli} = \epsilon$.

Assim, diferentemente de Li *et al.* (2021), Li *et al.* (2022) e Silveira Filho (2023), que utilizaram como critério de convergência a razão entre as normas euclidianas do vetor de forças residuais na i -ésima iteração, f_R^i , e o vetor de forças residuais da primeira iteração, f_R^0 , as Eqs. (3.1) e (3.2) utilizam a norma euclidiana do vetor de reações do modelo da mesoescala na iteração inicial, f_M^0 , como parâmetro para verificar a diferença relativa entre o vetor de forças residuais corrente e o carregamento do problema inicial. Em Li *et al.* (2022) ao aplicar o procedimento de solução *staggered*, tipos diferentes de modelos de elementos finitos eram utilizados na construção do modelo global e do modelo da mesoescala, além disso, parte dos fenômenos localizados eram representados na escala intermediária, o que garantia a diferença de rigidez inicial entre os modelos da escala meso e global. Nos exemplos numéricos que serão tratados neste trabalho, Seção 4, os efeitos localizados são representados apenas no modelo da escala local, e não há diferença

de rigidez inicial entre os modelos da escala meso e global, o que implicaria em um denominador f_R^0 muito pequeno. Tal fato motivou o uso do novo critério de convergência.

As principais ideias por trás do algoritmo *staggered* são, portanto, usar as iterações GLI, computacionalmente mais baratas, para melhorar a solução da mesoescala, e, após ocorrer a primeira simulação MEFG^{gl}, valer-se do modelo da mesoescala enriquecido nas iterações GLI subsequentes. A combinação dessas técnicas reduz o número de simulações MEFG^{gl} necessárias para atingir a convergência da estratégia, podendo fazer o Algoritmo 3.2 computacionalmente mais eficiente.

No algoritmo de solução monolítica, além das simulações MEFG^{gl}, computacionalmente mais caras, ocorrerem após as iterações GLI em cada iteração da estratégia, nas implementações do procedimento de solução monolítica propostas em Li *et al.* (2021) e Silveira Filho (2023), as iterações GLI ocorrem sempre entre o modelo da escala global e o modelo da mesoescala sem enriquecimento.

A necessidade de armazenar o enriquecimento do modelo da mesoescala para realizar as iterações GLI foi o que motivou a implementação do algoritmo de acoplamento no ambiente INSANE. Maneiras de exportar os dados de enriquecimento do modelo da mesoescala poderiam onerar o procedimento de análise e, dessa forma, eliminar a principal vantagem da abordagem de solução *staggered* em relação a abordagem monolítica.

Por fim, é importante destacar que, em simulações de problemas com propagação de trinca, o Algoritmo 3.2 é executado a cada passo de propagação, de modo que, a trinca só avança após a convergência das iterações *staggered*. Assim, o número de iterações *staggered* e de iterações GLI podem ser diferentes em cada passo de propagação.

3.2 Implementação Computacional

A implementação dos procedimentos de solução da estratégia não intrusiva MEFG^{gl}-GLI envolve três principais atores: o *software* comercial *Abaqus*, e as classes `IGLGlobalLocal` e `IGLGlobalLocalSimulationManager` implementadas no ambiente INSANE.

Em linhas gerais, a classe `IGLGlobalLocalSimulationManager` é a classe que implementa os Algoritmos 3.1 e 3.2, apresentados na Seção 3.1, e os métodos relacionados ao enriquecimento e solução dos modelos da mesoescala e da escala local; a classe `IGLGlobalLocal` é a classe que implementa os métodos relacionados ao acoplamento, além daqueles responsáveis pela comunicação com o *software* comercial; e, o *Abaqus* é a aplicação responsável pela solução do modelo da escala global. As Figs. 3.1 e 3.2 apresentam diagramas de sequência da implementação para os procedimentos de solução da abordagem monolítica e da abordagem *staggered*, respectivamente. Os métodos destacados nas Figs. 3.1 e 3.2 serão descritos com mais detalhes nas seções que seguem.

A comunicação da classe `IGLGlobalLocal`, implementada no ambiente INSANE, e o *Abaqus*, é realizada por intermédio de *scripts* desenvolvidos por Silveira Filho (2023)

Algoritmo 3.2: Solução *staggered* para o MEF G^{gl} -GLI.

Entrada:

- Modelos numéricos referentes as escalas global, meso e local;
- Lista de nós da interface entre os modelos global e meso;
- Lista de elementos da interface entre os modelos global e meso;
- Tipo de técnica de relaxação e coeficiente de relaxação (ω);
- Número de ciclos global-local (n_{GL});
- Tolerância do GLI (ϵ).

Saída:

- Solução final do modelo da escala global;
- Solução final do modelo da escala meso.

```

1   $i_s \leftarrow 0$ ;
2   $i_{gli} \leftarrow 0$ ;
3  Calcular a solução inicial do modelo global :  $\mathbf{u}_G^0$ ;
4  enquanto não convergir faça
5      enquanto não convergir faça
6          se  $i_{gli} > 0$  então
7               $\bar{\mathbf{u}}_G^i = \mathbf{u}_G^{i-1} + \mathbf{K}_G^{-1} \mathbf{f}_R^i$  ;                               /* solução prevista */
8              se relaxação dinâmica então
9                  Calcular  $\omega^i$  pela Eq. (2.12) ;                               /* atualização de  $\omega$  */
10             senão
11                  $\omega^i = \omega$  ;
12             fim
13              $\mathbf{u}_G^i = \mathbf{u}_G^{i-1} + \omega^i (\bar{\mathbf{u}}_G^i - \mathbf{u}_G^{i-1})$  ;                               /* solução corrigida */
14         senão
15              $\mathbf{u}_G^i = \bar{\mathbf{u}}_G^i$  ;                               /* não há correção na primeira iteração */
16         fim
17         Impor  $\mathbf{u}_G$  como condição de contorno em  $\Gamma_I$  na mesoescala;
18         Calcular a solução do modelo da mesoescala:  $\mathbf{u}_M^0$  (se  $i_s = 0$ ) ou  $\mathbf{u}_M^E$  (se  $i_s \neq 0$ );
19         Calcular  $\mathbf{f}_R$  pela Eq. (2.5) utilizando  $\mathbf{u}_G$ ,  $\mathbf{u}_M^0$  ou  $\mathbf{u}_M^E$ ;
20         se há convergência então
21             interromper;
22         fim
23          $i_{gli} \leftarrow i_{gli} + 1$ ;
24     fim
25     para  $j = 0$  até  $n_{GL}$  faça
26         Impor  $\mathbf{u}_M^0$  ou  $\mathbf{u}_{M,j}^E$  como condição de contorno do modelo local;
27         Calcular a solução do modelo local:  $\mathbf{u}_{L,j}$ ;
28         Enriquecer o modelo da mesoescala a partir da solução  $\mathbf{u}_{L,j}$ ;
29         Calcular a solução do modelo enriquecido da mesoescala:  $\mathbf{u}_{M,j}^E$ ;
30     fim
31     Calcular  $\mathbf{f}_R$  pela Eq. (2.5) utilizando  $\mathbf{u}_G$  e  $\mathbf{u}_{M,n_{GL}}^E$ ;
32     se há convergência então
33         interromper;
34     fim
35      $i_s \leftarrow i_s + 1$ ;
36 fim

```

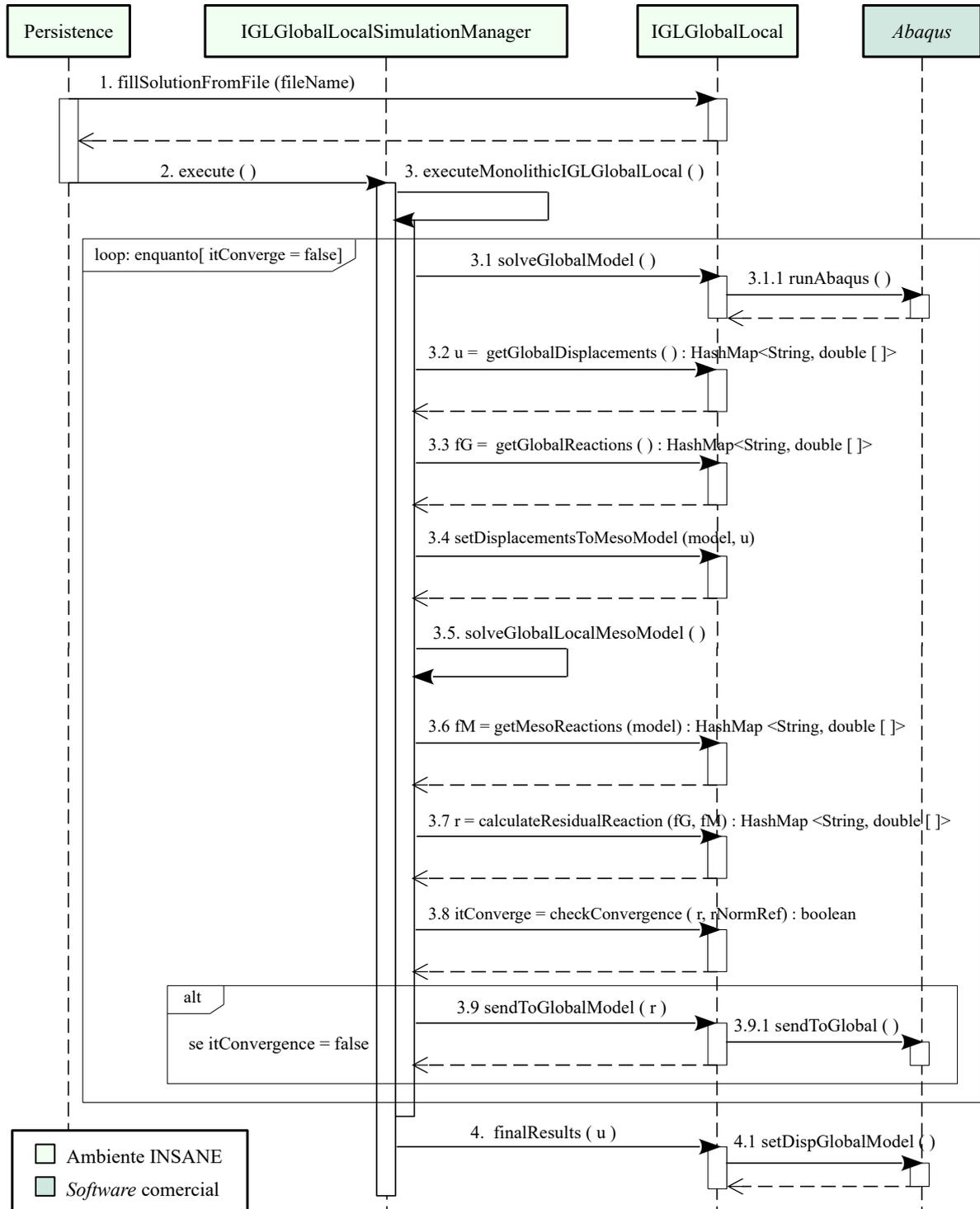


Figura 3.1: Diagrama de seqüência dos principais passos do procedimento de solução da abordagem monolítica.

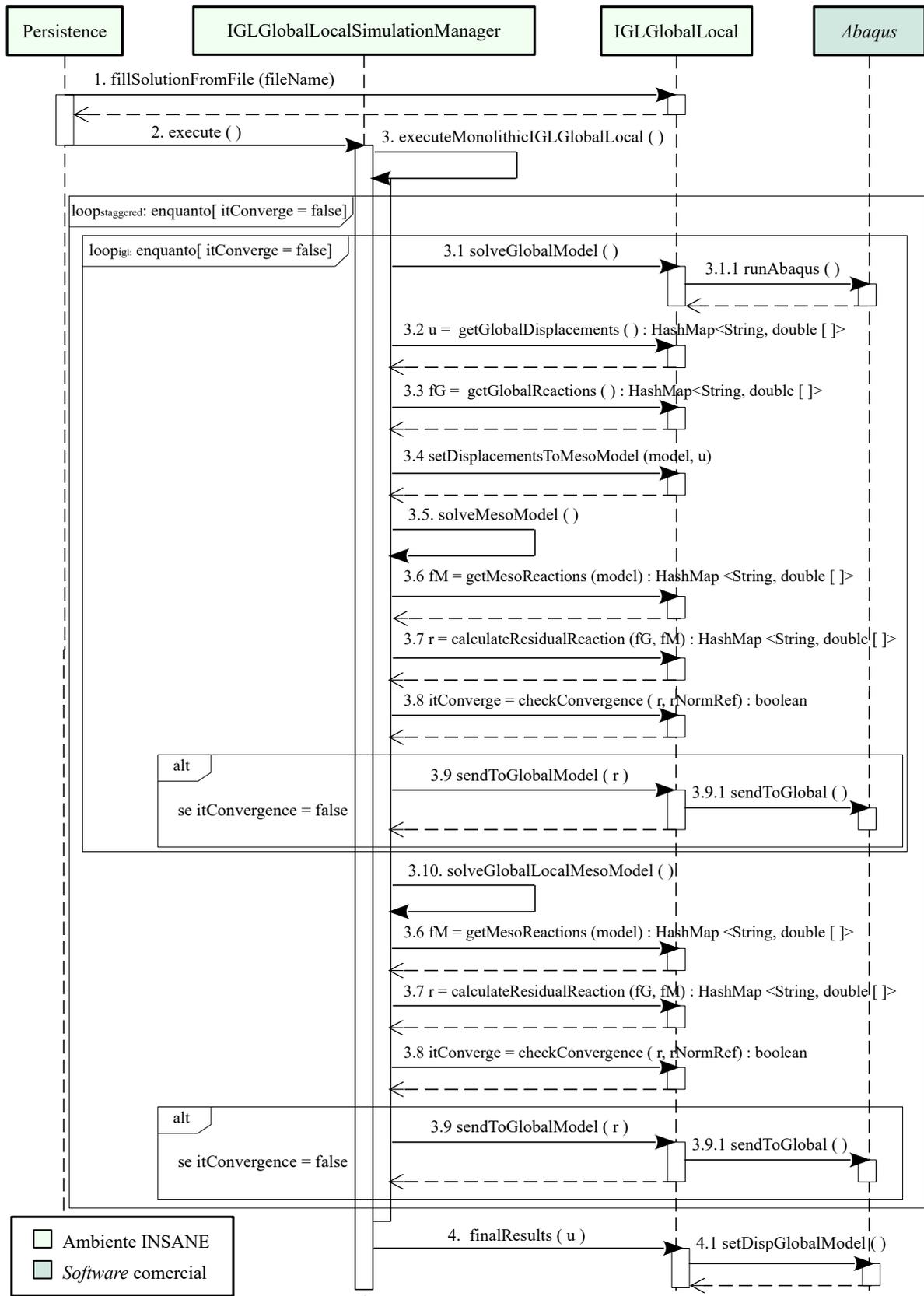


Figura 3.2: Diagrama de seqüência dos principais passos do procedimento de solução da abordagem *staggered*.

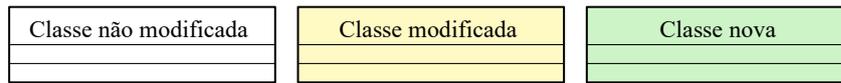


Figura 3.3: Notação UML utilizada nos diagramas de classes.

em linguagem de programação Python (Python, 2021). A execução desses *scripts* dentro do ambiente INSANE é realizada com o auxílio do projeto Jython (Jython, 2022). O Jython permite o desenvolvimento de aplicações híbridas que unem códigos em Java e Python, assim, através dele, foi possível criar um objeto do tipo `PythonInterpreter` que interpreta um código em Python dentro de um ambiente desenvolvido em Java. Os requisitos para execução da implementação são apresentados no Apêndice A.

As técnicas de relaxação, embora não indicadas nos diagramas de processos, são implementadas a partir da interface `IglAcceleration`. Essa classe tem como principal objetivo permitir a implementação das técnicas de relaxação sem a necessidade de alterações em outras partes do código.

Além da criação das classes `IGLGlobalLocalSimulationManager`, `IGLGlobalLocal` e `IglAcceleration`, a implementação computacional proposta neste trabalho incluiu a adaptação e utilização de classes já existentes no ambiente INSANE. Maiores detalhes sobre a implementação serão dados nas seções a seguir.

Diagramas UML (*Unified Modelling Language*) serão utilizados para melhor compreensão das modificações ou adições realizadas no código. A Fig. (3.3) apresenta a legenda adotada na descrição da implementação: classes não modificadas (em branco), classes alteradas (em amarelo) e classes criadas (em verde).

3.2.1 Entrada de dados

Além da entrada dos modelos numéricos das escalas global, meso e local, a implementação requer a entrada de parâmetros referentes ao acoplamento: tipo de procedimento de solução, tipo de técnica de relaxação, coeficiente de relaxação, tolerância do GLI, nós e elementos da interface do modelo global, nós e elementos da interface do modelo da mesoescala, e número de ciclos global-local.

Estes parâmetros são informados junto ao modelo numérico da mesoescala e, portanto, serão lidos pelo método `fillIGLGlobalLocalSolutionFromFile()` implementado na classe `PersistenceAsXml` (Fig. 3.4).

A classe `PersistenceAsXml` implementa a interface `Persistence`, que faz parte do núcleo de pré e pós-processamento do ambiente INSANE. Durante a leitura dos dados a interface atuará no pré-processamento das informações e, a partir dos dados de entrada, irá construir objetos do tipo `IGLGlobalLocalSimulationManager` e `IGLGlobalLocal`.

Detalhes sobre os arquivos para entrada de dados são apresentados no Apêndice B.

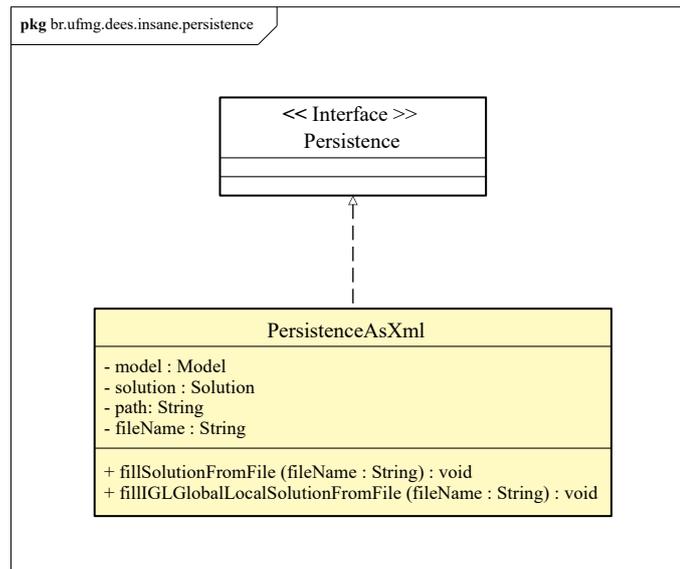


Figura 3.4: Modificações da classe `PersistenceAsXml`.

3.2.2 Classe `IGLGlobalLocal`

`IGLGlobalLocal` é uma classe do tipo `Solution`, filha da classe `GlobalLocal` (Fig. 3.5), e, herda, portanto, métodos e atributos úteis destas classes. Sua inicialização ocorre ainda na classe `PersistenceAsXml`, através do método `fillSolutionFromFile()`, que cria um objeto `Solution` do tipo `IGLGlobalLocal`. Os seus atributos também são preenchidos na classe `PersistenceAsXml`, por meio do método `fillIGLGlobalLocalSolutionFromFile()` mencionado na Seção 3.2.1 anterior. A criação do objeto `IGLGlobalLocal` inclui também a preparação dos arquivos para o acoplamento, criação de pastas para arquivos temporários e de saída de dados.

A classe `IGLGlobalLocal` é a classe responsável por conter os atributos e os métodos necessários para realizar o acoplamento entre os modelos da escala global e de mesoescala.

Assim, com exceção do atributo do tipo `IglAcceleration`, os demais atributos desta classe consistem em dados de entrada da implementação, e são descritos em detalhes no Apêndice B. O atributo `iglAcc` do tipo `IglAcceleration` é um objeto associado à técnica de relaxação utilizada no procedimento de solução, e será melhor detalhado na Seção 3.2.3.

Os principais métodos da classe `IGLGlobalLocal` são listados na Fig. 3.5, e serão descritos a seguir. Foram omitidos os métodos de importância secundária, relacionados à manipulações convenientes de atributos e gerenciamento dos arquivos de entrada e saída de dados.

- `solveGlobalModel()`: é o método que inicializa o procedimento de solução do modelo da escala global. Este método aciona o método `runAbaqus()`;

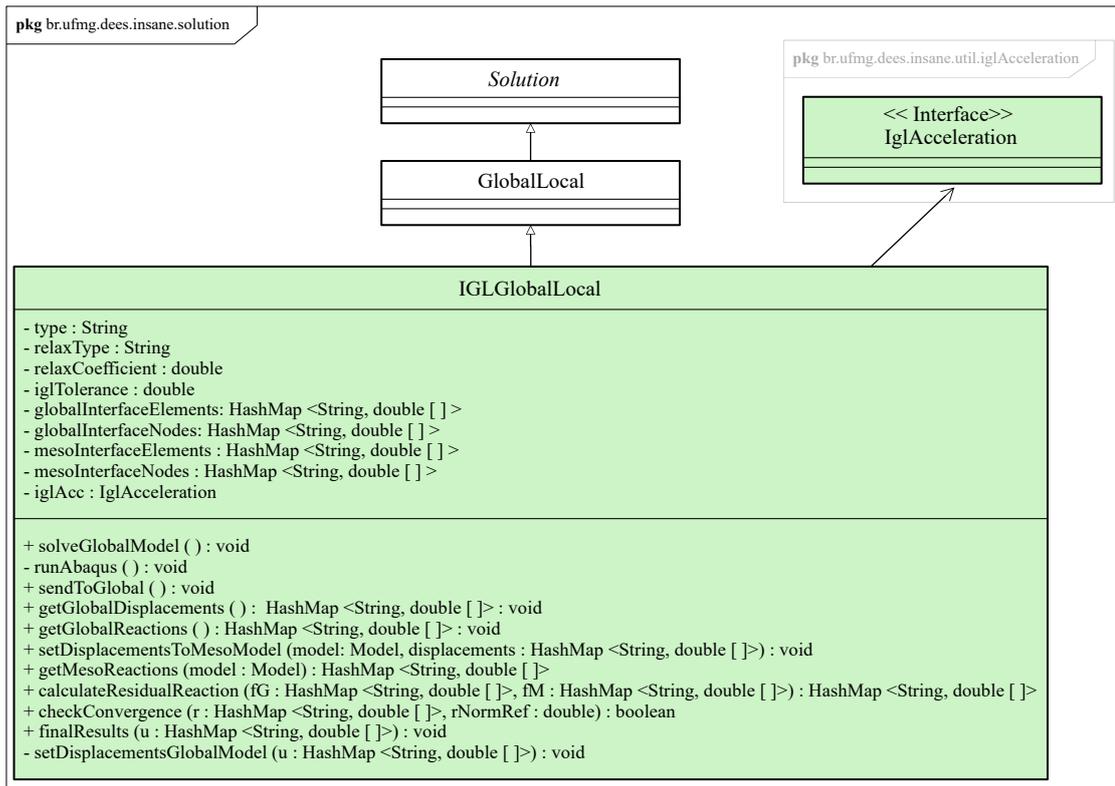


Figura 3.5: Criação da classe IGLGlobalLocal.

- `runAbaqus()`: soluciona o modelo global a partir da interpretação de *scripts* Python, e retorna arquivos de saída do tipo `.odb` e `.txt`;
- `sendToGlobal()`: envia forças nodais residuais para o modelo global a partir da interpretação de *scripts* Python;
- `getGlobalDisplacements()`: lê arquivo de saída `.txt` gerado no método `runAbaqus()` e retorna um objeto do tipo `HashMap` com os deslocamentos do modelo global;
- `getGlobalReactions()`: lê arquivo de saída `.txt` gerado no método `runAbaqus()` e retorna um objeto do tipo `HashMap` com as forças internas nos nós da interface do modelo global;
- `setDisplacementsToMesoModel()`: atribui aos nós da interface do modelo de mesoescala os deslocamentos prescritos;
- `getMesoReactions()`: retorna um objeto do tipo `HashMap` com as reações nos nós da interface do modelo da mesoescala;
- `calculateResidualReaction()`: calcula e retorna um objeto do tipo `HashMap` com as forças residuais para a iteração corrente calculadas conforme Eq. (2.5);
- `checkConvergence()`: verifica a convergência a partir da Eq. (3.1) ou (3.2), sendo `rNormRef` o valor calculado de $\|f_M^0\|$;
- `finalResults()`: aciona o método `setDisplacementsGlobalModel()` e soluciona o modelo global pela última vez;

- `setDisplacementsGlobalModel()`: atribui os deslocamentos acumulados aos nós do modelo global a partir da interpretação de scripts Python.

Observação 1. É importante esclarecer aqui o objetivo do método `finalResults()`. Para isso, deve-se lembrar que, como descrito na Seção 2.2.1, a implementação das técnicas de relaxação requer que a solução do problema global seja escrita na sua forma incremental. Sendo assim, da forma como a estratégia está implementada, o modelo da escala global é solucionado considerando o carregamento inicial apenas na primeira iteração. Nas iterações que seguem, o modelo da escala global é solucionado considerando apenas as forças residuais. Os deslocamentos são então acumulados durante o processo de solução e, só após a convergência da estratégia, eles são atribuído aos nós do modelo para se obter a solução final do modelo da escala global.

3.2.3 Classe `IglAcceleration`

A classe `IglAcceleration` é uma transcrição, seguido de uma adaptação, da classe `accelToolBox` desenvolvida por Silveira Filho (2023) em linguagem de programação Python.

`IglAcceleration` foi implementada aqui como uma interface, que tem por objetivo permitir e facilitar a implementação das técnicas de relaxação. Ela reúne os atributos e os métodos comuns às técnicas de relaxação implementadas.

Assim, a técnica de relaxação estática, apresentada na Seção 2.2.1.1, foi implementada na classe `StaticRelax`, enquanto a técnica de relaxação dinâmica, discutida na Seção 2.2.1.2, foi implementada na classe `DynamicRelax`. Ambas as classes implementam a interface `IglAcceleration` (Fig. 3.6).

Objetos das classes `StaticRelax` e `DynamicRelax` são criados a partir de um método *factory* implementado na interface `IglAcceleration`. Os demais métodos da interface são apenas assinaturas de métodos que devem ser implementados pelas classes que implementam `IglAcceleration`.

A construção das técnicas de relaxação precisa de dois atributos principais: o atributo `type` e o atributo `coefficient`, que correspondem à denominação do tipo de técnica de relaxação aplicada e ao fator de relaxação, respectivamente. Os dois atributos são dados de entrada da implementação, e, por isso, são passados como parâmetro a partir da classe `IGLGlobalLocal`.

A classe `StaticRelax` ainda conta com o atributo `uCorr`, que armazena a solução corrigida do passo anterior (\mathbf{u}_G^{i-1}) para realizar a correção do passo corrente (\mathbf{u}_G^i), conforme Eq. (2.9). Já a classe `DynamicRelax` conta com mais três atributos adicionais: `uCorr`, `uPrev` e `nodes`.

O atributo `nodes` da classe `DynamicRelax` corresponde ao nós da interface do modelo global. Já o atributo `uPrev` é um objeto do tipo `ArrayList` de duas posições, que são

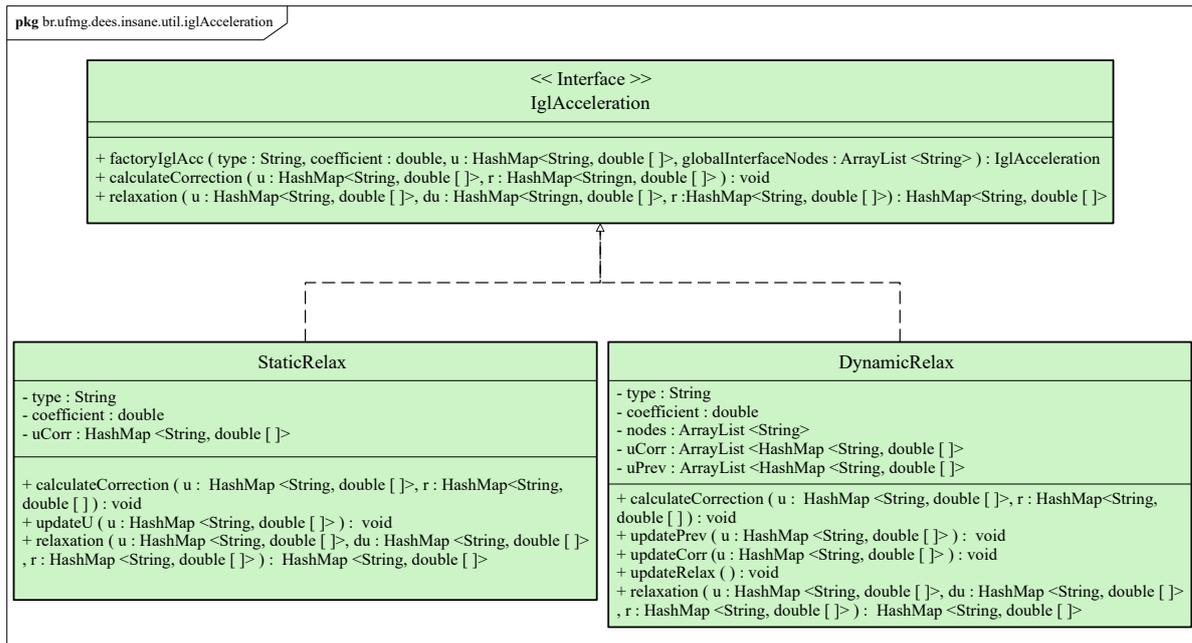


Figura 3.6: Criação da interface `IglAcceleration`.

inicialmente vazias. Com a execução do código, a primeira posição da lista é ocupada pela solução prevista no passo anterior ($\bar{\mathbf{u}}_G^{i-1}$), enquanto a segunda posição é ocupada pela solução prevista no antepenúltimo passo ($\bar{\mathbf{u}}_G^{i-2}$). O atributo `uCorr` funciona da mesma maneira, mas para o armazenamento das soluções corrigidas. Os atributos `uPrev`, `uCorr` e `nodes` são utilizados na atualização do fator de relaxação, ω , de acordo com a Eq. (2.12).

As classes `StaticRelax` e `DynamicRelax` devem obrigatoriamente implementar os métodos `relaxation()` e `calculateCorrection()`. O primeiro é o método responsável por realizar o processo de relaxação, enquanto, o segundo, tem por função realizar a correção da solução prevista. Além desses, são implementados, nas duas classes, métodos para atualizar o atributo `uCorr` a cada iteração. Na classe `DynamicRelax` são implementados ainda métodos para atualizar o atributo `uPrev` e o atributo referente ao fator de relaxação (`coefficient`).

3.2.4 Classe `IGLGlobalLocalSimulationManager`

A classe `IGLGlobalLocalSimulationManager` é a generalização da classe que implementa os algoritmos referentes aos procedimentos de solução da abordagem monolítica e *staggered* para a estratégia `MEFGgl-GLI`. Ela deve ser criada e implementada de acordo com as peculiaridades do problema que se queira solucionar através da estratégia de acoplamento. No contexto desta implementação, para o tratamento de problemas com propagação de trinca, a classe `IGLGlobalLocalSimulationManager` corresponde a classe `IGLGlobalLocalCrackPropagationSimulationManager`.

Assim, `IGLGlobalLocalCrackPropagationSimulationManager` é uma classe do tipo

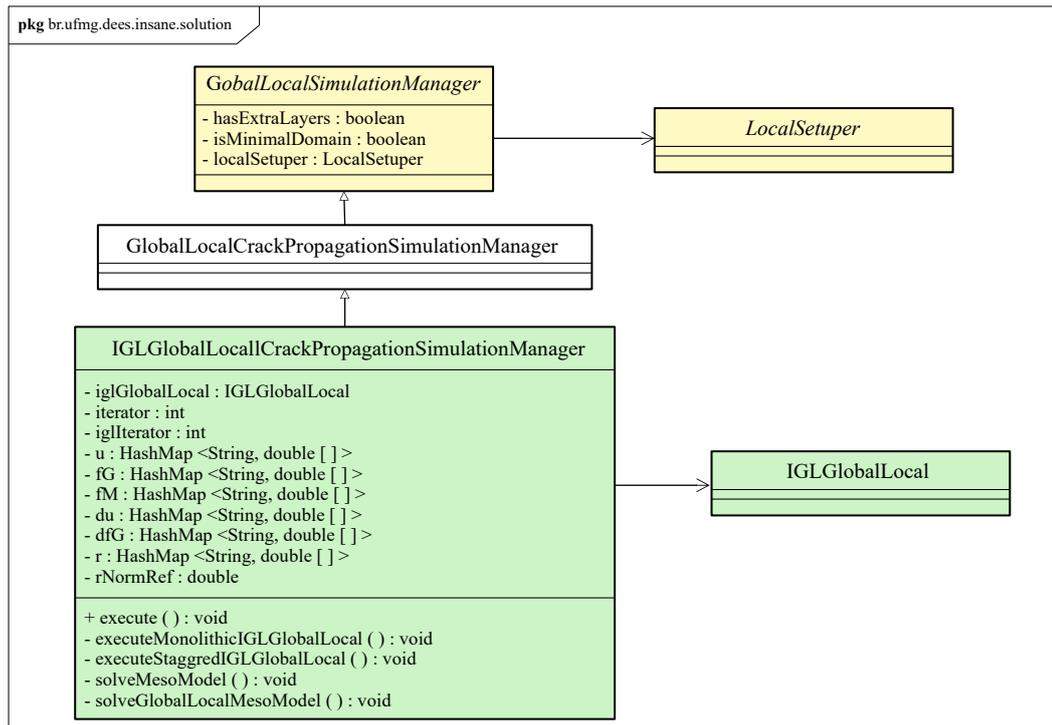


Figura 3.7: Criação da classe `IGLGlobalLocalCrackPropagationSimulationManager`.

`GlobalLocalSimulationManager` e que estende a classe `GlobalLocalCrackPropagationSimulationManager` (Fig. 3.7).

Os atributos da classe `IGLGlobalLocalCrackPropagationSimulationManager` são em grande parte variáveis utilizadas para receber e enviar informações da/para classe `IGLGlobalLocal` durante o procedimento de solução da estratégia de acoplamento. Desse modo, o atributo `iglGlobalLocal` é o objeto utilizado para acessar os atributos e os métodos da classe `IGLGlobalLocal`. Por isso, enquanto os demais atributos podem variar de acordo com a classe de solução proposta, o objeto do tipo `IGLGlobalLocal` deve ser um atributo comum a todas às classes que desejam implementar a estratégia de acoplamento.

Os principais atributos da classe `IGLGlobalLocalCrackPropagationSimulationManager` são listados e descritos na Tab. 3.1.

Os Algoritmos 3.1 e 3.2 são implementados, respectivamente, pelos métodos `executeMonolithicIGLGlobalLocal()` e `executeStaggeredIGLGlobalLocal()` da classe `IGLGlobalLocalCrackPropagationSimulationManager`, e são acionados pelo método `execute()` desta mesma classe. Assim, a depender do tipo de procedimento de solução informado no arquivo de entrada de dados, um dos dois métodos será executado.

Os demais métodos da classe `IGLGlobalLocalCrackPropagationSimulationManager` estão relacionados a solução e ao enriquecimento do modelo da mesoescala com funções global-local construídas a partir da solução do modelo da escala local. Estes métodos, em geral, utilizam ou adaptam outros métodos já existentes da classe mãe.

Assim, no contexto dessa implementação, a solução do MEF G^{gl} é realizada a partir

Tabela 3.1: Principais atributos da classe.

Atributo	Tipo	Descrição
<code>iglGlobalLocal</code>	<code>IGLGlobalLocal</code>	Objeto utilizado para acessar os atributos e métodos da classe <code>IGLGlobalLocal</code> ;
<code>iterator</code>	<code>int</code>	Variável que conta as iterações monolíticas e <i>staggered</i> ;
<code>iglIterator</code>	<code>int</code>	Variável que conta as iterações GLI;
<code>u</code>	<code>HashMap<String, double[]></code>	Variável que recebe e acumula os deslocamentos do modelo global;
<code>fG</code>	<code>HashMap<String, double[]></code>	Variável que recebe e acumula as reações nos nós da interface do modelo global;
<code>fM</code>	<code>HashMap<String, double[]></code>	Variável que recebe as reações nos nós da interface do modelo da mesoescala;
<code>du</code>	<code>HashMap<String, double[]></code>	Variável que recebe os deslocamentos do modelo global a cada iteração;
<code>dfG</code>	<code>HashMap<String, double[]></code>	Variável que recebe as reações nos nós da interface do modelo global a cada iteração;
<code>r</code>	<code>HashMap<String, double[]></code>	Variável que recebe as reações residuais;
<code>rNormRef</code>	<code>double</code>	Variável que corresponde ao $\ f_M^0\ $.

da construção dos modelos global e local pela classe `GlobalLocalSimulationManager`. Estes modelos correspondem aos modelos meso e local na estratégia de acoplamento, e são, portanto, construídos pelas classes do pacote `Setuper`.

A inclusão e propagação de trinca no modelo da escala local é realizada através de métodos implementados na classe `GlobalLocalCrackPropagationSimulationManager`, desenvolvida por Fonseca (2019). Nessa abordagem o modelo local é construído de forma automatizada, a partir da nuvem de elementos do modelo global (no caso do acoplamento, do modelo da mesoescala) que são interceptados pela trinca. Desse modo, tem-se sempre o menor modelo local possível.

Para viabilizar as verificações propostas no presente trabalho, modificações foram realizadas na classe `LocalSetuperGfem` (Fig. 3.8). Essa classe é a que constrói e configura o modelo local para análise via MEFG^{gl}. A modificação realizada consistiu na adição do método `getGlobalElementsForExtraLayers()` à classe, e implicou na adição dos atributos `hasExtraLayers` e `isMinimalDomain` em `GlobalLocalSimulationManager` (Fig. 3.7).

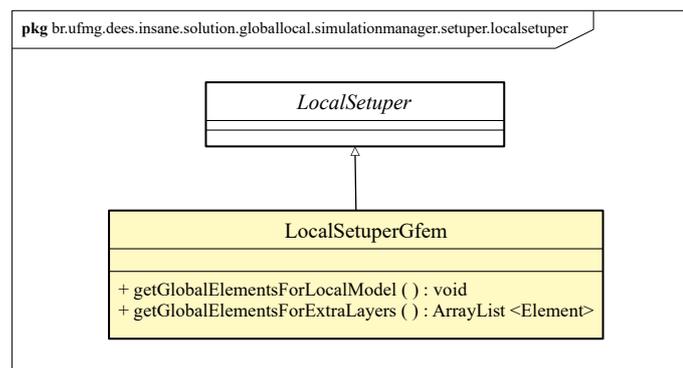


Figura 3.8: Modificação da classe `LocalSetuperGfem`.

O método adicionado é acionado dentro do método `getGlobalElementsForLocalModel()` já existente, e só é executado quando o atributo `hasExtraLayers` é de valor `true`. Este método adiciona camadas (*layers*) de elementos à seleção mínima de elementos do modelo global na construção do modelo local. O atributo `isMinimalDomain` sinaliza se o enriquecimento global-local dos nós do modelo global (modelo da mesoescala no caso do acoplamento) ocorrerá apenas nos nós cortados pela trinca, ou se ocorrerá em todos os nós possíveis do modelo. As modificações realizadas possibilitam verificações relacionadas a zona *buffer*, que não estavam, em plenitude, disponíveis na implementação de Fonseca (2019).

CAPÍTULO 4

SIMULAÇÕES NUMÉRICAS

Este capítulo apresenta os resultados de simulações numéricas para validar a implementação das estratégias de acoplamento não intrusivo MEFG^{gl}-GLI no ambiente INSANE. Os resultados obtidos são comparados com modelos de referência, construídos por malhas mais refinadas de elementos finitos ou de discretização equivalente à malha do modelo da escala local. Para cada exemplo numérico apresentado são realizadas investigações acerca de parâmetros relevantes da estratégia de solução: tipo de procedimento de solução, comportamento da convergência, e desempenho computacional. O desempenho computacional das estratégias MEFG^{gl}-GLI é avaliado em função do tempo computacional, fornecido pelo próprio ambiente INSANE. Assim, para as análises tratadas aqui, o tempo computacional corresponde a média do tempo computacional de três simulações consecutivas, realizadas em computador com processador Intel(R) Core(TM) i5-7200U (2.50-2.70GHz).

4.1 Chapa com orifício quadrado

O primeiro exemplo numérico apresentado baseia-se na simulação proposta por Silveira Filho (2023), e consiste em uma chapa tracionada com um orifício quadrado sob EPT. A única fonte de efeitos localizados é devido ao orifício situado na região central da chapa. O material é homogêneo e isotrópico, com módulo de elasticidade $E = 2,0 \times 10^{11}$, e coeficiente de Poisson $\nu = 0,30$. Na Fig. 4.1 apresenta-se o problema e suas dimensões. Unidades consistentes são utilizadas e serão omitidas neste texto.

Conforme proposto pela estratégia MEFG^{gl}-GLI, foram construídos três modelos computacionais distintos, todos discretizados com elementos finitos do tipo Q4 e ordem de integração 2×2 . As malhas dos modelos da escala global e da mesoescala possuem elementos com dimensão $l = 5,0$. Já a malha do modelo local, único modelo que representa o orifício, é construída com elementos com dimensão $l = 2,5$. As três escalas resultam em um total de 690 graus de liberdade. Foram construídos ainda mais dois modelos, Modelo A e Modelo B, obtidos a partir do *software* comercial *Abaqus* com solução via MEF em sua formulação convencional. Os modelos A e B também utilizam elementos finitos do tipo Q4, com dimensões $l = 2,5$ e $l = 0,625$, resultando em 1.064 e 15.680 graus de liberdade,

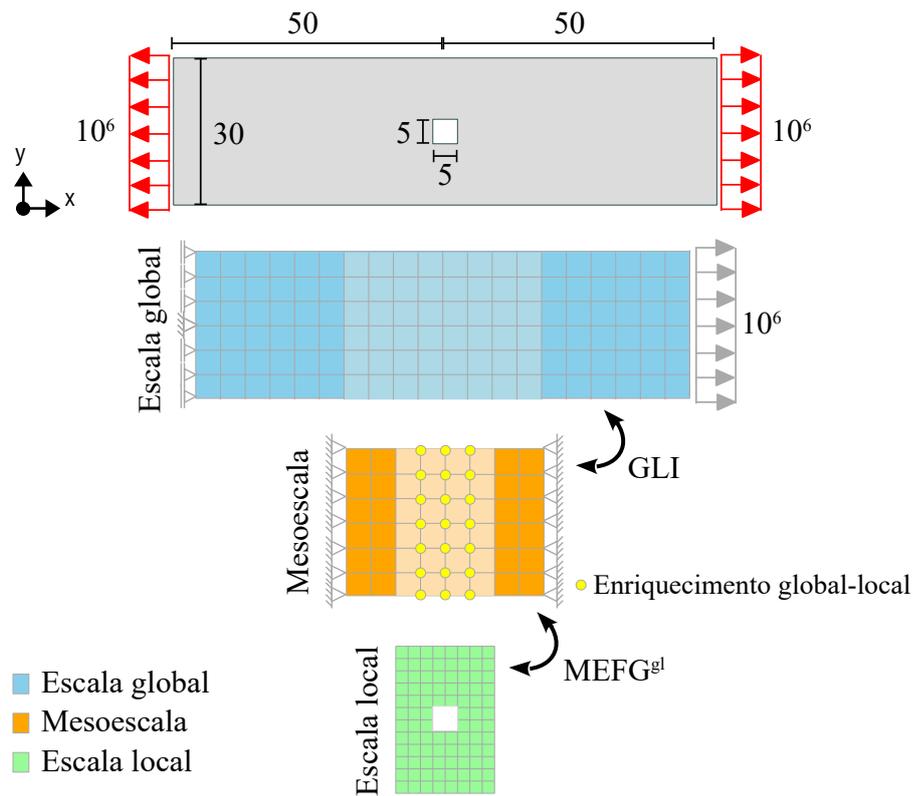


Figura 4.1: Chapa tracionada com orifício quadrado.

respectivamente.

O principal objetivo desse exemplo é validar a implementação do Algoritmo 3.2 de solução *staggered* da estratégia $\text{MEFG}^{gl}\text{-GLI}$ ($\text{MEFG}^{gl}\text{-GLI}^{St}$), comparando os resultados obtidos aqui com aqueles obtidos por Silveira Filho (2023), ao implementar o Algoritmo 3.1 de solução monolítica ($\text{MEFG}^{gl}\text{-GLI}^{M1}$), e, pela solução de modelos de elementos finitos de malha mais refinada, modelos A e B. Os resultados são avaliados em termos da componente de deslocamento na direção y , da componente de tensão normal na direção x , e do desempenho computacional.

No contexto da aplicação das estratégias de acoplamento, são considerados, exceto quando afirmado o contrário, um único ciclo de iteração global-local entre o modelos da mesoescala e da escala local ($n_{GL} = 1$), ausência de técnica de relaxação, e tolerância de $\epsilon_s = \epsilon_{gli} = 10^{-5}$, conforme Eq. (3.2). Nas simulações MEFG^{gl} das estratégias $\text{MEFG}^{gl}\text{-GLI}$ as informações transferidas do modelo da mesoescala para o modelo da escala local, são impostas como condições de contorno de Dirichlet (Seção 2.3.3), adotando valor pré-definido de $\bar{\kappa} = 2 \times 10^{10}$. De acordo com Kim *et al.* (2010), as condições de contorno de Dirichlet conduzem a resultados piores do que as condições de contorno de Cauchy, assim, as condições de contorno de Dirichlet são aplicadas nos limites do problema local a fim de verificar, para as simulações MEFG^{gl} , o pior cenário possível.

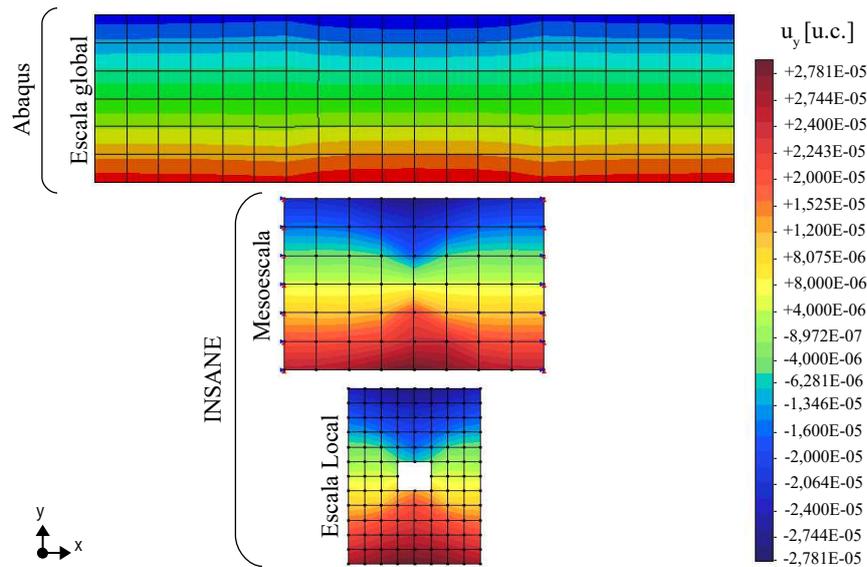


Figura 4.2: Deslocamentos na direção y obtidos através do $\text{MEFG}^{gl}\text{-GLI}^{St}$.

4.1.1 Deslocamento na direção y , u_y

Os resultados obtidos com a simulação realizada utilizando o algoritmo de solução *staggered* em termos da componente de deslocamento na direção y (u_y), são apresentados na Fig. 4.2. É possível notar que, ainda que só o modelo da escala local represente o orifício, o efeito do fenômeno localizado é levado para a escala intermediária sob a forma de funções de enriquecimento global-local, provocando perda de rigidez na região central do modelo da mesoescala. Nota-se ainda que, devido a transferência de forças e deslocamentos entre os modelos da escala global e intermediária, o modelo global tem sua solução corrigida, sob o seu domínio complementar, provocando um pequeno incremento no campo de deslocamentos próximo à região central da chapa. O resultado na região local do domínio global não deve ser utilizada, sendo substituída pela solução obtida na mesoescala. Essas observações explicitam o funcionamento da estratégia de acoplamento, e são bons indicativos da coerência da implementação.

Os deslocamentos na direção y obtidos pelos modelos A e B também são apresentados nas Figs. 4.3 e 4.4, respectivamente. O Modelo B é o modelo que possui maior número de graus de liberdade, e será, por isso, admitido como modelo de referência. Já o Modelo A, que tem a mesma discretização utilizada no problema local da simulação com acoplamento, se refere ao modelo equivalente. Observa-se assim, a partir das Figs. 4.2, 4.3 e 4.4, que os campos de deslocamento obtidos são de comportamento semelhante. Em todos os casos, os maiores deslocamentos verticais ocorrem, em módulo, nas extremidades inferiores e superiores da chapa.

Para melhor avaliar a qualidade dos resultados em termos de u_y , os deslocamentos verticais obtidos pelo $\text{MEFG}^{gl}\text{-GLI}^{St}$ ao longo da borda inferior da chapa foram comparados com os resultados obtidos pela implementação de Silveira Filho (2023) ($\text{MEFG}^{gl}\text{-GLI}^{M1}$),

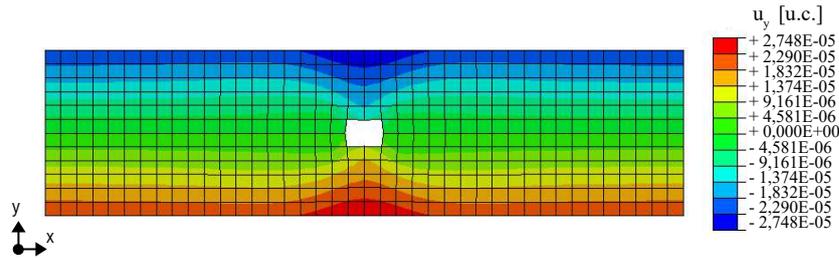


Figura 4.3: Deslocamentos na direção y obtidos pelo Modelo A.

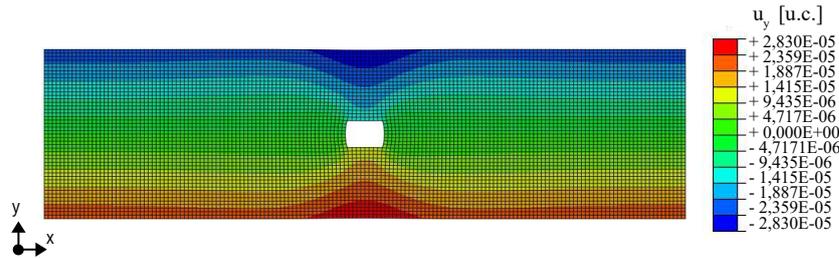


Figura 4.4: Deslocamentos na direção y obtidos pelo Modelo B.

e pelos modelos A e B, conforme Fig. 4.5. Nota-se então, que os resultados obtidos através do $\text{MEFG}^{gl}\text{-GLI}^{St}$ são coincidentes com os resultados obtidos pelo $\text{MEFG}^{gl}\text{-GLI}^{M1}$, e estão em conformidade com os resultados obtidos pelos modelos A e B. As soluções obtidas pelas estratégias $\text{MEFG}^{gl}\text{-GLI}$ podem se apresentar, portanto, como soluções alternativas à solução obtida pelo Modelo A, uma vez que se aproximam da solução de referência (Modelo B) utilizando menor número de graus de liberdade.

Os resultados em termos de u_y apresentados na Fig. 4.5 indicam uma melhor aproximação das soluções obtidas através das estratégias $\text{MEFG}^{gl}\text{-GLI}$ em relação a solução de referência (Modelo B) quando comparadas à solução equivalente (Modelo A). Não seria prudente, porém, afirmar que em termos globais o acoplamento está de fato melhorando a solução com relação à equivalente. Mesmo porque, essa conclusão não é esperada, pois se trata de uma discretização mais pobre. Essa afirmação requer a comparação entre as energias de deformação, que é um parâmetro que melhor reflete o comportamento global das soluções. Assim, as energias de deformação calculadas para as soluções das estratégias $\text{MEFG}^{gl}\text{-GLI}$, do Modelo A e do Modelo B foram, respectivamente, de $7,6638 \times 10^3$ u.c., $7,6696 \times 10^3$ u.c., e $7,6939 \times 10^3$ u.c. . Logo, a solução do Modelo B é a que possui maior energia de deformação, devido ao maior número de graus de liberdade. Ainda que próximas, a energia de deformação da solução do Modelo A é maior do que a energia de deformação das estratégias $\text{MEFG}^{gl}\text{-GLI}$, também devido ao maior número de graus de liberdade do Modelo A, que, diferentemente das estratégias de acoplamento, utiliza discretização mais refinada em todo o domínio do problema.

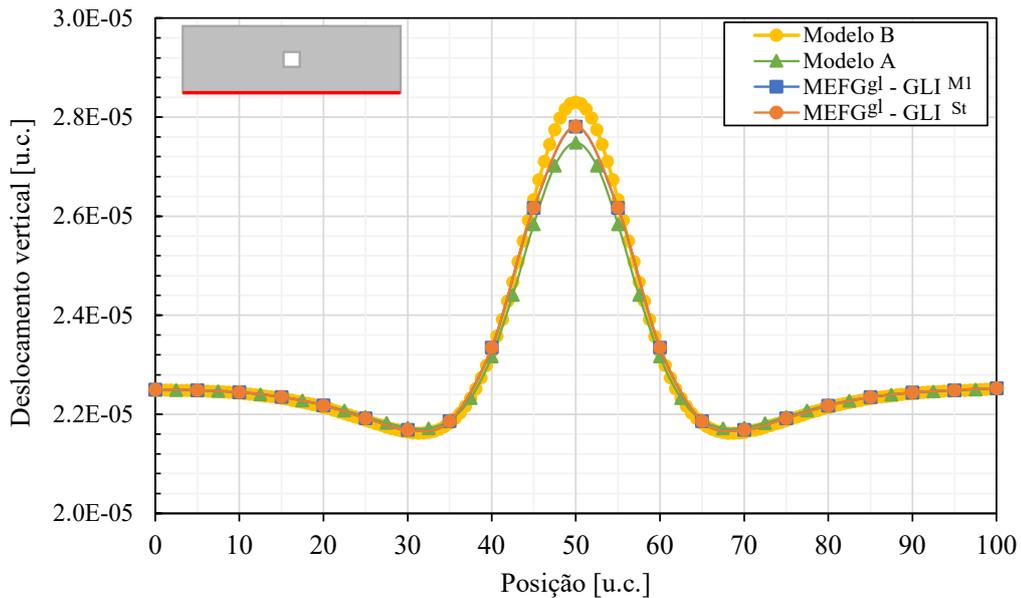


Figura 4.5: Comparação dos resultados em termos de u_y ao longo da borda inferior da chapa obtidos pelo MEFG^{gl}-GLISt, pelo MEFG^{gl}-GLI^{M1}, e pelos modelos A e B.

4.1.2 Componente de tensão normal na direção x , σ_{xx}

Semelhante à Fig. 4.2, os resultados obtidos com o algoritmo de solução *staggered* em termos da componente de tensão normal na direção x (σ_{xx}) são apresentados na Fig. 4.6. As tensões σ_{xx} obtidas pelos modelos A e B também são apresentadas. Neste caso é importante destacar que as escalas de cores para representação dos resultados são definidas de forma distinta pelo *Abaqus* e pelo *INSANE*, o que implica em maior cautela na análise dos resultados. Além disso, é necessário ressaltar, que o resultado na região local do domínio global deve ser desprezado, sendo substituído pela solução obtida na mesoescala.

Adicionalmente, deve-se lembrar que o orifício é representado apenas no modelo da escala local, e sua posição cobre 1/4 dos quatro elementos centrais do modelo da mesoescala. Assim, como o orifício não é geometricamente representado no modelo da mesoescala, sua presença é indiretamente considerada pelo enriquecimento fornecido pela solução numérica local. Além disso, nos quatro elementos centrais do modelo da mesoescala, a matriz de rigidez é calculada pela contribuição dos pontos numéricos de integração dos elementos locais. Contudo, ainda que nenhuma integração seja realizada na região do furo, a aproximação associada ao nó central do modelo da mesoescala, coincidente com o orifício, contribui para a solução. A consequência é suavizar a solução local descontínua na aproximação da mesoescala. A representação gráfica agrava esse fato, já que utiliza um campo de tensão recuperado para extrapolar os resultados associados aos nós de cada elemento, e, posteriormente, obtém as componentes das tensões nodais a partir dos valores médios resultantes da extrapolação nos elementos que compartilham o respectivo nó.

Os procedimentos de suavização, fornecidos pela solução global-local, e pela repre-

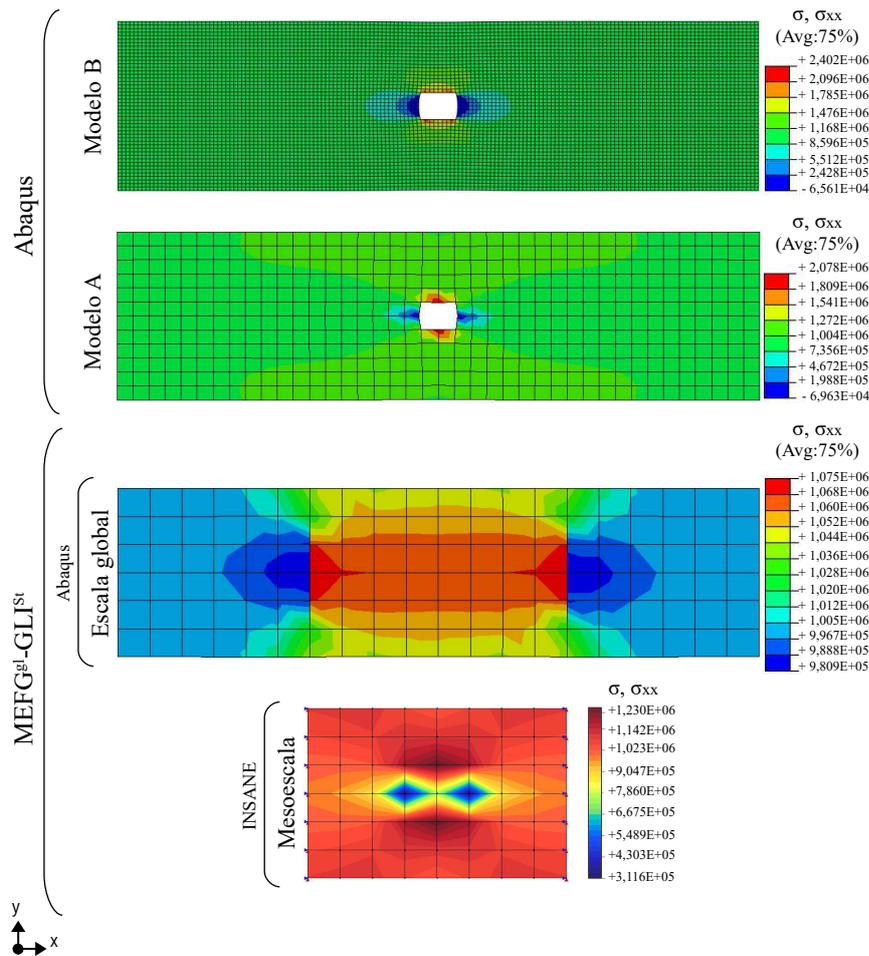


Figura 4.6: Tensões na direção x obtidos através do $\text{MEFG}^{gl}\text{-GLI}^{St}$, e pelos modelos A e B.

sentação gráfica, explicam as diferenças entre os valores extremos apresentados na Fig. 4.6 para os modelos A e B (obtidos pelo *Abaqus*) e o modelo da mesoescala (obtido pelo *INSANE*). Sem a suavização imposta pela representação gráfica, os valores máximo e mínimo obtidos diretamente da solução do modelo da mesoescala são, respectivamente, $2,065 \times 10^6$ e $-7,038 \times 10^4$, e estão de fato próximos dos valores obtidos pelo Modelo A (solução equivalente), onde o orifício é representado geometricamente.

Para avaliar com maior precisão a qualidade dos resultados obtidos com a implementação do algoritmo *staggered*, foram definidos três pontos de análise, conforme Fig. 4.7. Assim, os valores de tensão σ_{xx} fornecidos pela implementação foram pontualmente comparados com os resultados obtidos pelos modelos A e B.

No contexto da estratégia $\text{MEFG}^{gl}\text{-GLI}$, os pontos 1 e 2 correspondem ao domínio da mesoescala, enquanto que o ponto 3 está localizado no domínio complementar do modelo da escala global. Os pontos 1 e 2 foram igualmente definidos em Silveira Filho (2023) na avaliação dos resultados da estratégia $\text{MEFG}^{gl}\text{-GLI}^{M1}$ em termos das componentes de deformação, já o ponto 3 foi adicionalmente definido aqui para verificar o comportamento da solução fora do domínio da mesoescala. Ademais, os pontos definidos se referem a

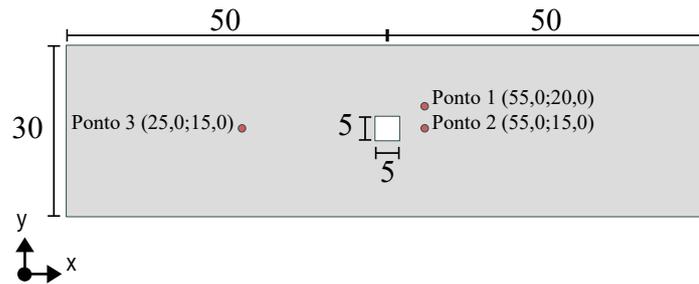


Figura 4.7: Definição dos pontos para comparação dos resultados da chapa.

Tabela 4.1: Comparação dos resultados em termos de σ_{xx} obtidos pelo $\text{MEFG}^{gl}\text{-GLI}^{St}$, e pelos modelos A e B.

Solução	Componente de tensão na direção x		
	Ponto 1	Ponto 2	Ponto 3
$\text{MEFG}^{gl}\text{-GLI}^{St}$	$1,1555 \times 10^6$	$3,1165 \times 10^5$	$9,8504 \times 10^5$
Modelo A	$1,1582 \times 10^6$	$3,1548 \times 10^5$	$9,8546 \times 10^5$
Modelo B	$1,1705 \times 10^6$	$3,3250 \times 10^5$	$9,8494 \times 10^5$

pontos nodais nas malhas das três soluções, sendo assim, o valor da tensão é dado pela média das tensões nodais dos elementos que compartilham o respectivo nó. A Tab. 4.1 compara os resultados em termos da componente de tensão σ_{xx} nos pontos destacados na Fig. 4.7.

Verifica-se, a partir da Tab. 4.1, que os resultados obtidos pela estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$, em termos da componente de tensão σ_{xx} , apresentam diferença de 1,28%, 6,27% e 0,01% em relação aos valores obtidos pelo Modelo B (de referência), para os pontos 1, 2 e 3, respectivamente. Já o Modelo A, apresenta diferença de 1,05% no ponto 1, 5,12% no ponto 2 e 0,05% no ponto 3, também em relação ao Modelo B. Ao comparar os resultados do $\text{MEFG}^{gl}\text{-GLI}^{St}$ com os valores obtidos pelo Modelo A verifica-se que a maior diferença acontece no Ponto 2, e é de apenas 1,21%.

O Modelo A tem discretização equivalente ao do modelo da escala local da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$, mas ao longo de todo o seu domínio, e, por isso, possui maior número de graus de liberdade (1.064 contra 690). Contudo, como já verificado anteriormente, mesmo com menos graus de liberdade, a solução da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ é capaz de se aproximar da solução do Modelo A. Em termos de custo computacional, o algoritmo implementado também pode ser vantajoso, uma vez que pode requerer menor tempo computacional do que a solução obtida pela implementação da estratégia $\text{MEFG}^{gl}\text{-GLI}^{M1}$ em Silveira Filho (2023), conforme mostrado na próxima Seção.

4.1.3 Desempenho Computacional

Para comparar a eficiência computacional da implementação do algoritmo *staggered* em relação ao algoritmo monolítico, analisou-se a solução do mesmo problema da Fig. 4.1 utilizando o procedimento de solução monolítica para o $\text{MEFG}^{gl}\text{-GLI}$ (Algoritmo 3.1) implementado no ambiente INSANE, indicada aqui por $\text{MEFG}^{gl}\text{-GLI}^{M2}$. Duas são as diferenças entre a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M1}$, implementada por Silveira Filho (2023), e a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$. Em $\text{MEFG}^{gl}\text{-GLI}^{M1}$ há uma aplicação em Python que gerencia as análises entre o *Abaqus* e o INSANE. Como consequência, em cada iteração GLI da estratégia o modelo da mesoescala corresponde ao modelo da mesoescala inicial, sem enriquecimento, exatamente como foi proposto em Li *et al.* (2021).

Já em $\text{MEFG}^{gl}\text{-GLI}^{M2}$, o gerenciamento do acoplamento é todo realizado a partir do INSANE. Por isso, diferentemente do que foi inicialmente proposto por Li *et al.* (2021), após a primeira iteração GLI, o modelo da mesoescala é atualizado, e passa a ser um modelo enriquecido. Tal modificação não interferiu de forma relevante nos resultados obtidos pelas duas soluções. Embora se tenha verificado que a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$ converge com uma iteração a menos do que a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M1}$, os resultados obtidos pelas duas estratégias, em termos de deslocamento vertical ao longo da borda inferior da chapa, foram praticamente idênticos.

A Fig. 4.8 apresenta a comparação do comportamento da convergência das estratégias $\text{MEFG}^{gl}\text{-GLI}^{St}$ e $\text{MEFG}^{gl}\text{-GLI}^{M2}$. É importante destacar aqui que, em ambos os casos, o contador das iterações é iniciado em zero e é incrementado ao longo da solução. No caso da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$, que possui dois iteradores (iterador *staggered* e iterador GLI), o iterador representado no gráfico corresponde ao contador das iterações GLI, para garantir equivalência às estas mesmas iterações contabilizadas para a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$. Deve-se lembrar ainda, que na implementação proposta no presente trabalho, a convergência das estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$ é verificada a partir do resíduo relativo dado pelas Eqs. (3.1) e (3.2), respectivamente. Sendo assim, como no problema apresentado na Fig. 4.1 os modelos da escala global e meso possuem a mesma discretização e mesmo módulo de elasticidade, a estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ converge já na iteração GLI inicial, ou seja, na iteração 0.

Verifica-se ainda, a partir da Fig. 4.8, que a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$ possui convergência linear, e converge com uma iteração a menos que a estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$. Nesse caso, a estratégia que implementa o algoritmo monolítico também requer cerca de 3,70% menos tempo computacional do que a estratégia que implementa o algoritmo *staggered*. Assim, ainda que a utilização do algoritmo *staggered* resulte em um menor número de simulações MEFG^{gl} (ocorrem apenas nas iterações 0, 4 e 5) e, consequentemente, em um menor número de ciclos global-local (3 ciclos contra 5 ciclos), a iteração GLI adicional compromete a eficiência da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ em relação a $\text{MEFG}^{gl}\text{-GLI}^{M2}$. É

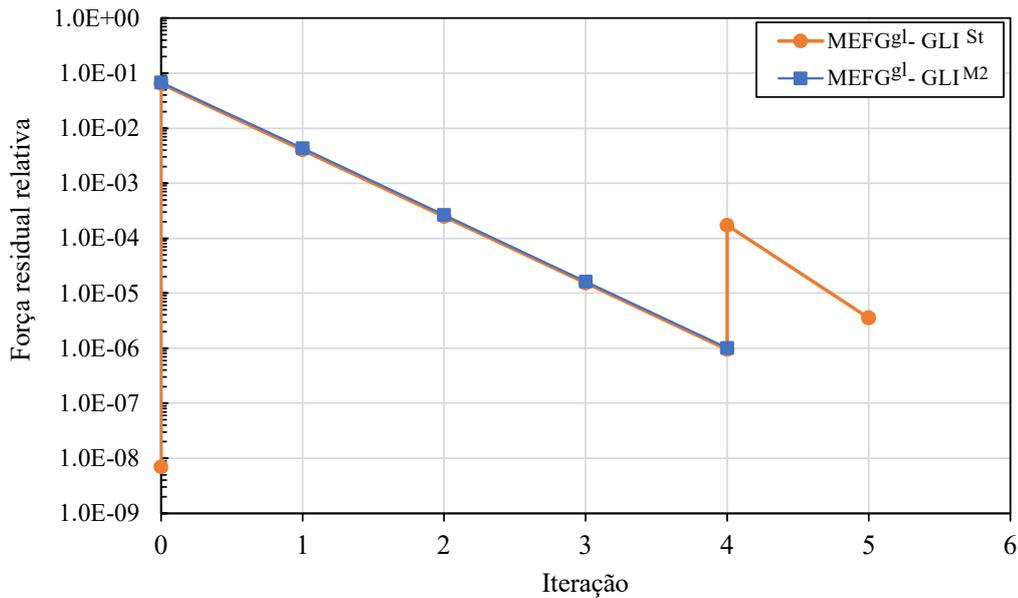


Figura 4.8: Comparação do comportamento da convergência das estratégias $\text{MEFG}^{gl}\text{-GLI}^{St}$ e $\text{MEFG}^{gl}\text{-GLI}^{M2}$ com $n_{GL} = 1$.

importante destacar também que, como no problema da Fig. 4.1, os modelos da escala global e meso não possuem diferença de rigidez relevante, a iteração GLI inicial da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ poderia ser ignorada e, dessa forma, ter seu custo computacional reduzido. Em um problema onde os modelos das escala global e meso possuem diferença de rigidez inicial – devido a diferenças de propriedades do material, presença de enrijecedores, ou inclusões, por exemplo – não haveria convergência na iteração GLI inicial, e portanto, essa observação não seria válida.

Com o objetivo de verificar a eficiência computacional das estratégias de acoplamento, com o aumento do número de ciclos global-local das simulações MEFG^{gl} , realizou-se a solução do mesmo problema da Fig. 4.1 adotando dois ciclos de iteração global-local. A Fig. 4.9 apresenta a comparação do comportamento da convergência das estratégias $\text{MEFG}^{gl}\text{-GLI}^{St}$ e $\text{MEFG}^{gl}\text{-GLI}^{M2}$ para $n_{GL} = 2$. Observa-se, a partir da Fig. 4.9, que a estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ converge com o mesmo número de iterações que a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$. Contudo, aqui, a estratégia que implementa o algoritmo *staggered* requer menos tempo computacional do que estratégia que implementa o algoritmo monolítico. Há uma redução de cerca de 19,80% do tempo computacional, devido, principalmente, ao menor número de ciclos global-local das simulações MEFG^{gl} realizadas pela estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$. Enquanto em $\text{MEFG}^{gl}\text{-GLI}^{M2}$, simulações MEFG^{gl} ocorrem a cada iteração da estratégia, resultando em dez ciclos de iteração global-local totais, em $\text{MEFG}^{gl}\text{-GLI}^{St}$ simulações MEFG^{gl} ocorrem somente na primeira e na última iteração (iterações 0 e 4), resultando em um total de quatro ciclos de iteração global-local.

É importante observar ainda, que, na estratégia que implementa o algoritmo monolítico, o aumento do número de ciclos global-local não interfere no número de iterações

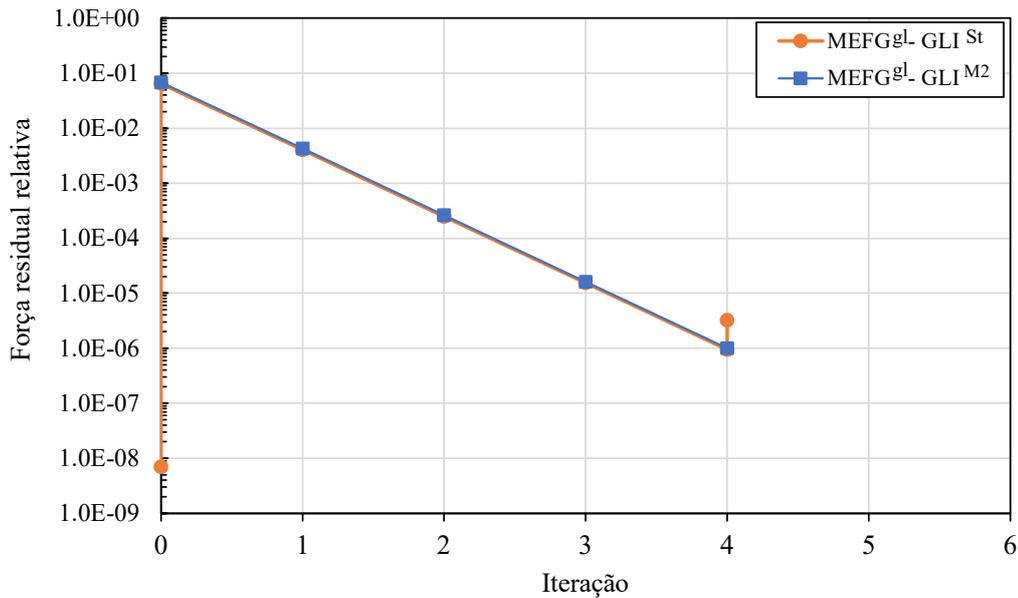


Figura 4.9: Comparação do comportamento da convergência das estratégias $\text{MEFG}^{gl}\text{-GLI}^{St}$ e $\text{MEFG}^{gl}\text{-GLI}^{M2}$ com $n_{GL} = 2$.

da estratégia, mas onera o procedimento de solução, aumentando o tempo computacional da análise em 17,83%. Já na estratégia que implementa o algoritmo *staggered*, o aumento do número de ciclos global-local reduz o número de iterações da estratégia, de 5 para 4, diminuindo o custo, e por consequência, o tempo computacional da solução em cerca de 8,86%. A redução do número de iterações da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ com o aumento do número de ciclos global-local é devido à melhora na qualidade da solução do modelo da mesoescala enriquecida, que acelera a convergência da estratégia sem comprometer a precisão dos resultados.

De modo geral verifica-se que, quanto maior o custo computacional associado às simulações MEFG^{gl} (maior número dos ciclos de iteração global-local, maior número de pontos de integração, maior número de efeitos localizados, etc.), mais relevante pode ser a eficiência da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ em relação a $\text{MEFG}^{gl}\text{-GLI}^{M2}$.

Nesse sentido, com o objetivo de verificar o comportamento das estratégias $\text{MEFG}^{gl}\text{-GLI}^{St}$ e $\text{MEFG}^{gl}\text{-GLI}^{M2}$ mediante ao aumento do esforço computacional das simulações MEFG^{gl} , foi realizado o enriquecimento dos nós dos vértices do orifício quadrado representado no modelo da escala local do problema apresentado na Fig. 4.1, mantendo $n_{GL} = 2$. O enriquecimento dos nós do modelo local foi acompanhado pelo aumento do número de pontos de integração dos elementos que circundam o orifício, para ser capaz de representar a aproximação mais complexa resultante. A Fig. 4.10 identifica os nós enriquecidos pelas funções de singularidade e os elementos do modelo da escala local que tiveram a ordem dos pontos de integração aumentados de 2×2 para 10×10 . As funções de singularidade foram descritas conforme as Eqs. (2.18), (2.19), (2.20) e (2.21), propostas por Szabo e Babuška (1991), considerando $\alpha = 270^\circ$, e, por consequência, assumindo $\lambda^{(1)} = 0,5444$,

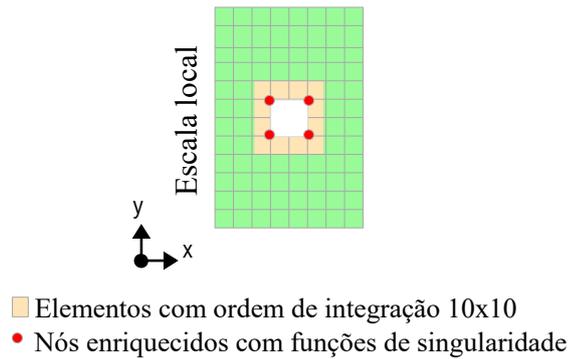


Figura 4.10: Identificação dos nós enriquecidos com funções de singularidade.

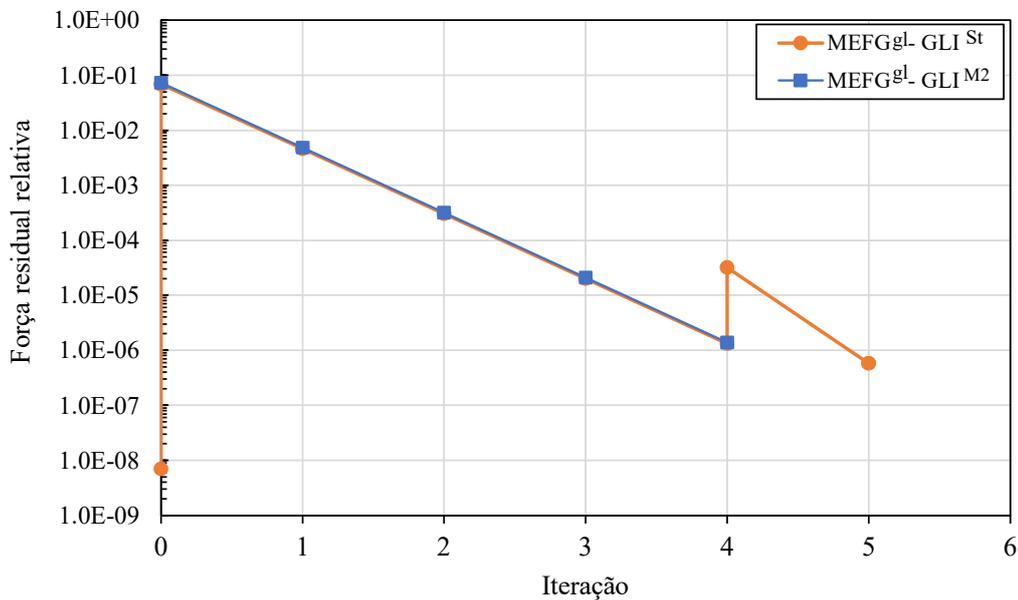


Figura 4.11: Comparação do comportamento da convergência das estratégias MEFG^{gl}-GLISt e MEFG^{gl}-GLI^{M2} com enriquecimento do modelo da escala local e $n_{GL} = 2$.

$\lambda^{(2)} = 0,9085$, $Q^{(1)} = 0,5431$, e $Q^{(2)} = -0,2189$.

A Fig. 4.11 apresenta a comparação do comportamento da convergência das estratégias MEFG^{gl}-GLISt e MEFG^{gl}-GLI^{M2} com enriquecimento dos nós do modelo da escala local. Verificou-se nesse caso que, embora a estratégia MEFG^{gl}-GLISt convirja com uma iteração a mais do que a estratégia MEFG^{gl}-GLI^{M2}, ela reduz o tempo computacional do procedimento de solução em cerca de 24,09%. Assim, como esperado, a utilização da estratégia MEFG^{gl}-GLISt resultou aqui, em uma redução de tempo computacional mais significativa do que a obtida na análise da Fig. 4.9. Como agora as simulações MEFG^{gl} estão computacionalmente mais onerosas e ocorrem em cada iteração do algoritmo de solução monolítica, elas têm maior impacto sobre o custo computacional da estratégia MEFG^{gl}-GLI^{M2} do que da estratégia MEFG^{gl}-GLISt.

Embora o enriquecimento dos nós do modelo da escala local com funções de singu-

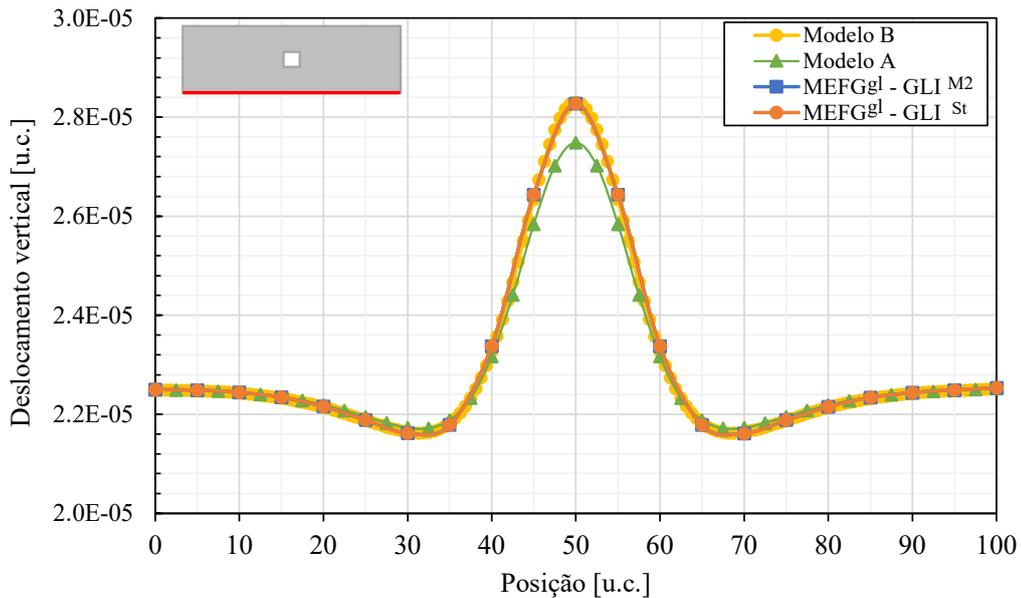


Figura 4.12: Comparação dos resultados em termos de u_y ao longo da borda inferior da chapa obtidos pelo $\text{MEFG}^{gl}\text{-GLI}^{St}$ e pelo $\text{MEFG}^{gl}\text{-GLI}^{M2}$ com enriquecimento do modelo da escala local, e pelos modelos A e B.

laridade onere o procedimento de solução das estratégias $\text{MEFG}^{gl}\text{-GLI}$, há também uma melhora na descrição do comportamento do campo de tensões nos vértices do orifício, onde estão localizados pontos de concentração de tensão. Deste modo, a solução passa a ser capaz de capturar com maior precisão os efeitos localizados, e, por consequência, melhorar a aproximação. A Fig. 4.12 apresenta essa verificação ao comparar os resultados em termos de deslocamento na direção y (u_y) na borda inferior da chapa da Fig. 4.1, obtidos pelas estratégias $\text{MEFG}^{gl}\text{-GLI}^{St}$ e $\text{MEFG}^{gl}\text{-GLI}^{M2}$, considerando enriquecimento do modelo da escala local, e pelos modelos A e B.

Assim, ao comparar os resultados apresentados na Fig. 4.12, onde ocorre o enriquecimento dos nós dos vértices do orifício do modelo da escala local, com aqueles apresentados na Fig. 4.5, no qual o enriquecimento não é realizado, verifica-se que há significativa melhora na aproximação das soluções das estratégias $\text{MEFG}^{gl}\text{-GLI}$ em relação ao Modelo B, de referência. Isso significa que, mesmo com uma discretização mais grosseira, o enriquecimento dos nós do modelo da escala local das estratégias $\text{MEFG}^{gl}\text{-GLI}$ com funções de singularidade são capazes de descrever os efeitos localizados nos vértices do orifício com precisão equivalente à obtida pela malha do Modelo B, que possui malha com elementos quatro vezes menores do que a malha do modelo da escala local.

As verificações realizadas até aqui, validam a implementação do algoritmo de solução *staggered* da estratégia $\text{MEFG}^{gl}\text{-GLI}$ para o tratamento de problemas elásticos lineares com a inclusão de vazios no modelo da escala local. No exemplo numérico a seguir os algoritmos de solução da estratégia $\text{MEFG}^{gl}\text{-GLI}$ são aplicados na resolução de um problema da MFLE.

4.2 Pórtico em “L” com propagação de trinca

O exemplo numérico consiste em um pórtico em “L” com carregamento de $2,1 \times 10^4$ aplicado na extremidade à direita, e inclusão de uma trinca inicial na região de singularidade. O material é homogêneo e isotrópico, com módulo de elasticidade $E = 2,0 \times 10^7$ e coeficiente de Poisson $\nu = 0,30$. A Fig. 4.13 apresenta o problema e suas dimensões. Unidades consistentes são utilizadas e serão omitidas neste texto.

Como proposto pela estratégia $\text{MEFG}^{\text{gl}}\text{-GLI}$, são construídos três modelos computacionais distintos, todos discretizados com elementos finitos do tipo Q4. Os elementos do modelo da escala global e do modelo da mesoescala possuem ordem de integração 2×2 . Já na escala local, a ordem de integração depende da posição do elemento em relação a trinca: os elementos que contêm a ponta da trinca ou que são interceptados por ela, possuem ordem de integração 12×12 , os demais elementos têm ordem de integração 4×4 . As malhas dos modelos da escala global e da mesoescala possuem elementos com dimensão $l = 5,0$. Já a malha do modelo da escala local é definida pela divisão das arestas dos elementos do modelo da mesoescala por três, resultando em elementos com dimensão $l = 1,6667$. A trinca, representada apenas no modelo da escala local, é definida por $a = 7,9057$ e $\beta = 2,8198$ rad.

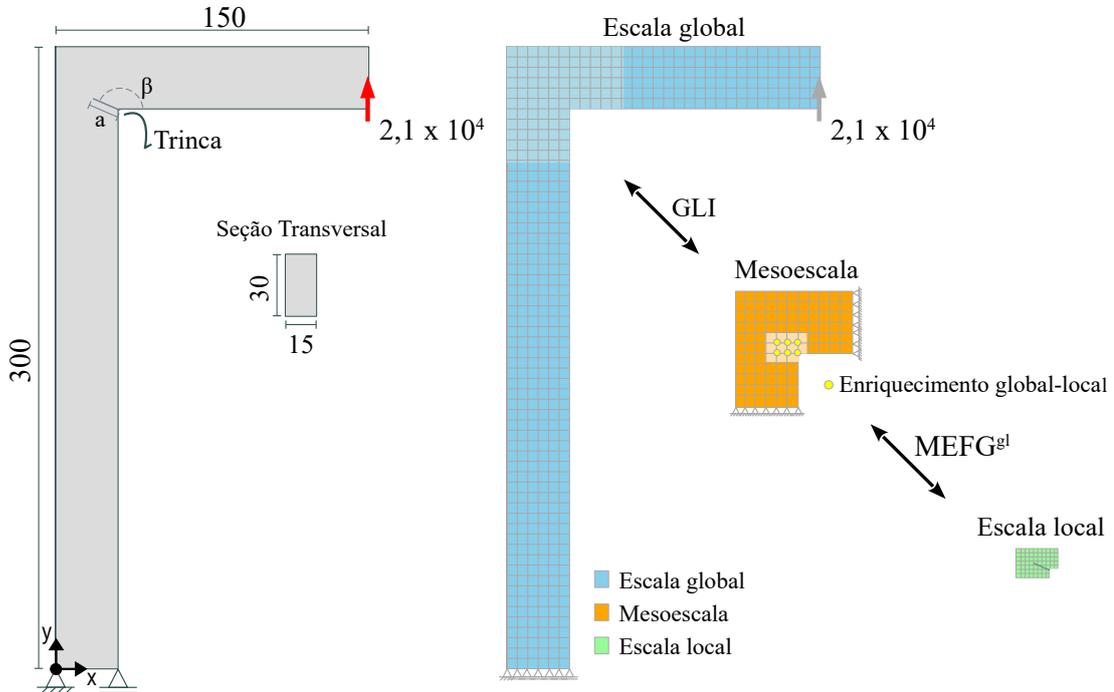


Figura 4.13: Pórtico em “L” com a inclusão de uma trinca.

É importante destacar que o modelo da escala local representado na Fig. 4.13 é construído automaticamente durante o procedimento de solução e, por isso, pode ter suas dimensões atualizadas durante o processo. Na simulação com propagação de trinca, por

exemplo, o modelo da escala local é atualizado a cada passo de propagação, influenciando, por consequência, no enriquecimento dos nós do modelo da mesoescala. Assim, as ilustrações da Fig. 4.13 têm maior relevância didática, e correspondem apenas ao problema da trinca estática, ou seja, sem propagação.

O objetivo desse exemplo é validar a implementação dos algoritmos de solução monolítica e *staggered* da estratégia $MEFG^{gl}$ -GLI no ambiente INSANE para o tratamento de problemas da MFLE, bem como avaliar a sensibilidade do acoplamento às variações dos parâmetros da simulação. Para isso, os resultados obtidos pelas estratégias $MEFG^{gl}$ -GLI são comparados com os resultados obtidos por um modelo numérico equivalente. Tal modelo é solucionado no ambiente INSANE via $MEFG^{gl}$ a partir do desenvolvimento de Fonseca (2019), assumindo problemas global e local com discretizações equivalentes às dos modelos da escala global e da escala local das estratégias $MEFG^{gl}$ -GLI, respectivamente.

No contexto das simulações $MEFG^{gl}$, realizadas tanto na solução do modelo numérico equivalente, quanto pelas estratégias de acoplamento, as informações transferidas do modelo da escala global (modelo da mesoescala, no caso do acoplamento) para o modelo da escala local, são impostas como condições de contorno de Cauchy (Seção 2.3.3). No presente trabalho, a definição das condições de contorno de Cauchy é realizada a partir do cálculo automatizado do valor de $\bar{\kappa}$, durante o processo de solução do problema local, por um método que implementa o procedimento de Birner e Schweitzer (2019) e utiliza um campo de tensões recuperadas como proposto por Fonseca (2023).

O problema representado na Fig. 4.13 é solucionado pelo modelo numérico equivalente e pelas estratégias $MEFG^{gl}$ -GLI considerando sete passos de propagação de trinca, com incrementos, Δa , de 1,6667. A validação da implementação das estratégias de acoplamento é realizada a partir de investigações de parâmetros relacionados ao $MEFG^{gl}$ (número de ciclos global-local, e tamanho da zona *buffer*). Em todos os casos foi considerado tolerância de $\epsilon_s = \epsilon_{gli} = 10^{-5}$, conforme Eqs. (3.1) e (3.2).

Para as investigações propostas, os resultados obtidos pelas estratégias $MEFG^{gl}$ -GLI são comparados com a solução de referência do modelo equivalente em termos dos fatores de intensidade de tensão, K_I e K_{II} , dos deslocamentos na direção y na extremidade direita do pórtico, e do desempenho computacional.

4.2.1 Solução de referência no modelo equivalente

Antes de prosseguir para a validação das estratégias de acoplamento, é necessário definir a solução de referência do modelo numérico equivalente. Ou seja, definir a solução que será utilizada para verificar a precisão e conformidade dos resultados obtidos pela implementação das estratégias $MEFG^{gl}$ -GLI.

Assim, com o objetivo de se obter a solução mais adequada para o problema apresentado na Fig. 4.13, será estudado aqui a influência do número de ciclos global-local e o

tamanho da zona *buffer* sobre a convergência da solução do modelo numérico equivalente. Como mencionado na Seção 2.3.4, os ciclos global-local e a zona *buffer* são parâmetros relacionados ao MEFG^{gl} que melhoram a qualidade das condições de contorno aplicadas ao problema local e, por isso, influenciam na precisão da solução do método.

A verificação da convergência da solução é realizada em termos dos fatores de intensidade de tensão (K_I e K_{II}) para a trinca estática (1° passo) e ao final do último passo de propagação da trinca (7° passo). A influência do número de ciclos global-local sobre a convergência da solução é realizada assumindo, inicialmente, domínio local mínimo, ou seja, ausência de zona *buffer*, e considerando um, dois, três, quatro e cinco ciclos global-local. As Tabs. 4.2 e 4.3 apresentam os resultados em termos dos fatores de intensidade de tensão obtidos no 1° e 7° passo de propagação, respectivamente, pelo modelo numérico equivalente em função do aumento do número de ciclos global-local. Os erros correspondem a diferença relativa entre os fatores de intensidade de tensão obtidos em um dado número de ciclos global-local em relação aos fatores de intensidade de tensão obtidos pelo número de ciclos global-local anterior. A Fig. 4.14 ilustra o problema global enriquecido no 1° e no 7° passo de propagação, onde, para esse caso, o domínio local coincide com a região de enriquecimento.

Tabela 4.2: Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente no 1° passo de propagação da trinca.

N° de ciclos	K_I	K_{II}	<i>Erro</i> K_I	<i>Erro</i> K_{II}
1	4062,7755	-381,8456	1,0000	1,0000
2	4056,4189	-383,1655	$1,5670 \times 10^{-3}$	$3,4447 \times 10^{-3}$
3	4056,5857	-383,2140	$4,1101 \times 10^{-5}$	$1,2668 \times 10^{-4}$
4	4056,5787	-383,2124	$1,7034 \times 10^{-6}$	$4,2438 \times 10^{-6}$
5	4056,5791	-383,2126	$8,1255 \times 10^{-8}$	$4,1615 \times 10^{-7}$

Tabela 4.3: Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente no 7° passo de propagação da trinca.

N° de ciclos	K_I	K_{II}	<i>Erro</i> K_I	<i>Erro</i> K_{II}
1	2503,3519	-194,1483	1,0000	1,0000
2	2483,3699	-201,9692	$8,0463 \times 10^{-3}$	$3,8724 \times 10^{-2}$
3	2485,8153	-200,2347	$9,8376 \times 10^{-4}$	$8,6628 \times 10^{-3}$
4	2485,5473	-200,3505	$1,0781 \times 10^{-4}$	$5,7792 \times 10^{-4}$
5	2485,5854	-200,3359	$1,5325 \times 10^{-5}$	$7,2553 \times 10^{-5}$

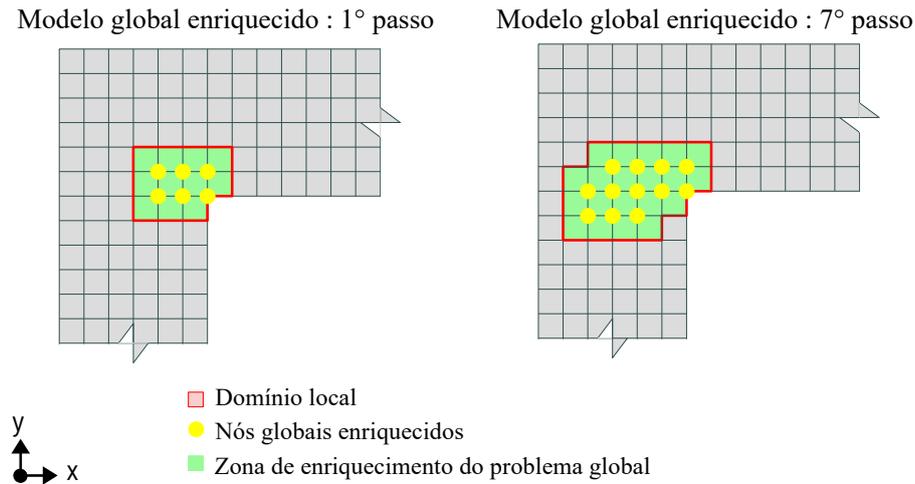


Figura 4.14: Problema global enriquecido no 1° e no 7° passo de propagação com ausência de zona *buffer*.

Verifica-se, a partir das Tabs. 4.2 e 4.3 que a convergência dos fatores de intensidade de tensão ocorrem mais rapidamente, em função do aumento do número de ciclos-global, no 1° passo de trinca do que no 7° passo. Observa-se ainda, que a convergência de K_{II} é mais sensível ao número de ciclos global-local, sendo ligeiramente mais lenta do que a convergência de K_I . A Fig. 4.15 ilustra essas observações. De qualquer forma, tanto no 1° quanto no 7° passo de propagação da trinca, mais do que três ciclos global-local não provocam alterações significativas nos resultados obtidos em termos dos fatores de intensidade de tensão, indicando a convergência da solução.

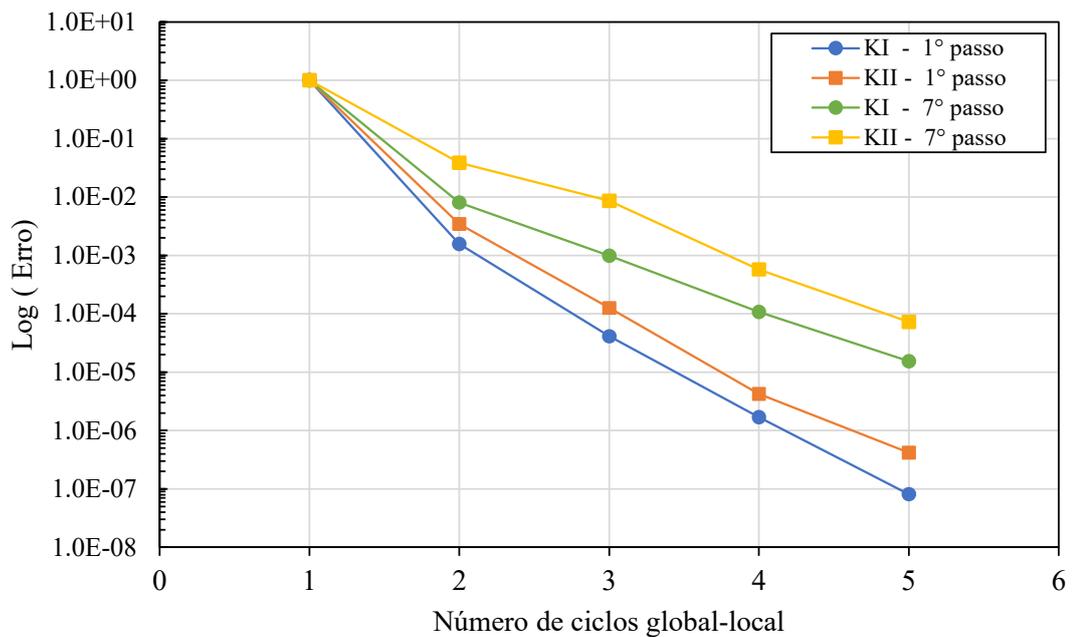


Figura 4.15: Comportamento da convergência de K_I e K_{II} em função do número de ciclos global-local no 1° e no 7° passo de propagação assumindo domínio local mínimo.

Para se avaliar melhor a acurácia da solução, torna-se importante considerar a influência da zona *buffer* na qualidade das soluções transferidas como condições de contorno. Sendo assim, também é realizada a verificação da influência do número de ciclos global-local sob a convergência da solução assumindo zona *buffer* com uma e duas camadas elementos. As soluções que envolvem zona *buffer* utilizam o método desenvolvido no presente trabalho, apresentado na Fig. 3.8, que adiciona camadas de elementos à seleção mínima de elementos do modelo global na construção do modelo local, mantendo o enriquecimento global-local apenas nos nós cortados pela trinca.

As Tabs. 4.4 e 4.5 apresentam os resultados em termos dos fatores de intensidade de tensão obtidos, no 1° e 7° passo de propagação, respectivamente, pelo modelo numérico equivalente com zona *buffer* de uma camada de elementos, em função do aumento do número de ciclos global-local. A Fig. 4.16 ilustra o problema global enriquecido no 1° e no 7° passo de propagação, com zona *buffer* de uma camada de elementos.

Tabela 4.4: Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente com zona *buffer* de uma camada de elementos no 1° passo de propagação da trinca.

N° de ciclos	K_I	K_{II}	$Erro K_I$	$Erro K_{II}$
1	4064,7308	-383,5558	1,0000	1,0000
2	4065,2363	-381,5555	$1,2435 \times 10^{-4}$	$5,2424 \times 10^{-3}$
3	4065,2262	-381,6194	$2,4952 \times 10^{-6}$	$1,6746 \times 10^{-4}$
4	4065,2263	-381,6168	$3,1661 \times 10^{-8}$	$6,8401 \times 10^{-6}$
5	4065,2263	-381,6169	$2,7720 \times 10^{-10}$	$2,8301 \times 10^{-7}$

Tabela 4.5: Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente com zona *buffer* de uma camada de elementos no 7° passo de propagação da trinca.

N° de ciclos	K_I	K_{II}	$Erro K_I$	$Erro K_{II}$
1	2473,7636	-202,2386	1,0000	1,0000
2	2466,5222	-208,6841	$2,9358 \times 10^{-3}$	$3,0886 \times 10^{-2}$
3	2467,0793	-208,5831	$2,2581 \times 10^{-4}$	$4,8393 \times 10^{-4}$
4	2467,0366	-208,5861	$1,7329 \times 10^{-5}$	$1,4337 \times 10^{-5}$
5	2467,0399	-208,5868	$1,3609 \times 10^{-6}$	$3,4892 \times 10^{-6}$

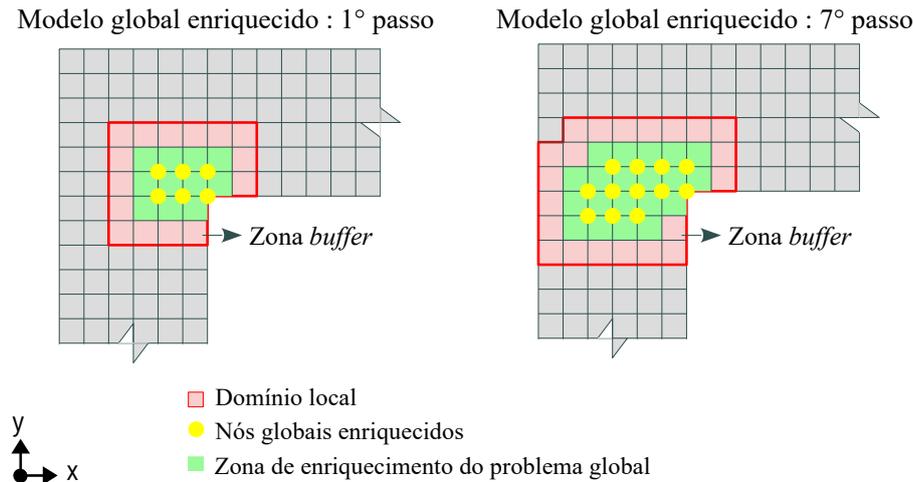


Figura 4.16: Problema global enriquecido no 1º e no 7º passo de propagação com zona *buffer* de uma camada de elementos.

Observa-se, a partir das Tabs. 4.4 e 4.5, que, para esse caso, mais do que dois ciclos de iteração global-local não alteram de forma relevante os resultados da solução em termos dos fatores de intensidade de tensão. Verifica-se, sobretudo, que os fatores de intensidade de tensão obtidos, no 1º e 7º passo de propagação da trinca, pelo modelo numérico equivalente com zona *buffer* de uma camada de elementos, convergem para valores diferentes daqueles obtidos pela solução do modelo numérico equivalente com domínio local mínimo (Tabs. 4.2 e 4.3). Tal fato dificulta a determinação da solução de referência, e justifica a solução do modelo numérico equivalente considerando zona *buffer* com duas camadas de elementos.

As Tabs. 4.6 e 4.7 apresentam os resultados em termos dos fatores de intensidade de tensão obtidos, no 1º e 7º passo de propagação, respectivamente, pelo modelo numérico equivalente com zona *buffer* de duas camadas de elementos, em função do aumento do número de ciclos global-local. A Fig. 4.17 ilustra o problema global enriquecido no 1º e no 7º passo de propagação, com zona *buffer* de duas camadas de elementos.

Verifica-se, que os resultados em termos dos fatores de intensidade de tensão obtidos, no 1º e 7º passo de propagação da trinca, pelo modelo numérico equivalente com zona *buffer* de uma camada de elementos (Tabs. 4.4 e 4.5), convergem para valores próximos dos apresentados nas Tabs. 4.6 e 4.7, obtidos pelo modelo numérico equivalente com zona *buffer* de duas camadas. Ainda que tal observação não garanta a precisão das soluções apresentadas nas Tabs. 4.6 e 4.7, é um bom indicativo da convergência da solução.

Além do mais, observa-se que, os resultados obtidos com ausência de zona *buffer*, com uma e duas camadas de elementos, apesar de apresentarem uma pequena diferença inicial em termos dos fatores de intensidade de tensão, conduzem a erros cada vez menores com o aumento do número de ciclos global-local. Os resultados em termos de K_I e K_{II} , no 7º passo de propagação da trinca, em função do aumento do número de ciclos global-local

e o aumento do tamanho da zona *buffer*, são exibidos, respectivamente, nas Figs. 4.18 e 4.19, e explicitam tal comportamento.

Tabela 4.6: Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente com zona *buffer* de duas camadas de elementos no 1° passo de propagação da trinca.

N° de ciclos	K_I	K_{II}	$Erro K_I$	$Erro K_{II}$
1	4064,7743	-381,9758	1,0000	1,0000
2	4065,0888	-381,5138	$7,7368 \times 10^{-5}$	$1,2110 \times 10^{-3}$
3	4065,0876	-381,5178	$2,9180 \times 10^{-7}$	$1,0367 \times 10^{-5}$
4	4065,0876	-381,5177	$2,1405 \times 10^{-10}$	$1,6737 \times 10^{-7}$
5	4065,0876	-381,5177	$3,3311 \times 10^{-11}$	$2,8249 \times 10^{-9}$

Tabela 4.7: Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente com zona *buffer* de duas camadas de elementos no 7° passo de propagação da trinca.

N° de ciclos	K_I	K_{II}	$Erro K_I$	$Erro K_{II}$
1	2468,1028	-202,4830	1,0000	1,0000
2	2468,6958	-203,8736	$2,4024 \times 10^{-4}$	$6,8206 \times 10^{-3}$
3	2468,7783	-203,9130	$3,3395 \times 10^{-5}$	$1,9338 \times 10^{-4}$
4	2468,7754	-203,9107	$1,1523 \times 10^{-6}$	$1,1087 \times 10^{-5}$
5	2468,7755	-203,9109	$4,0236 \times 10^{-8}$	$5,6629 \times 10^{-7}$

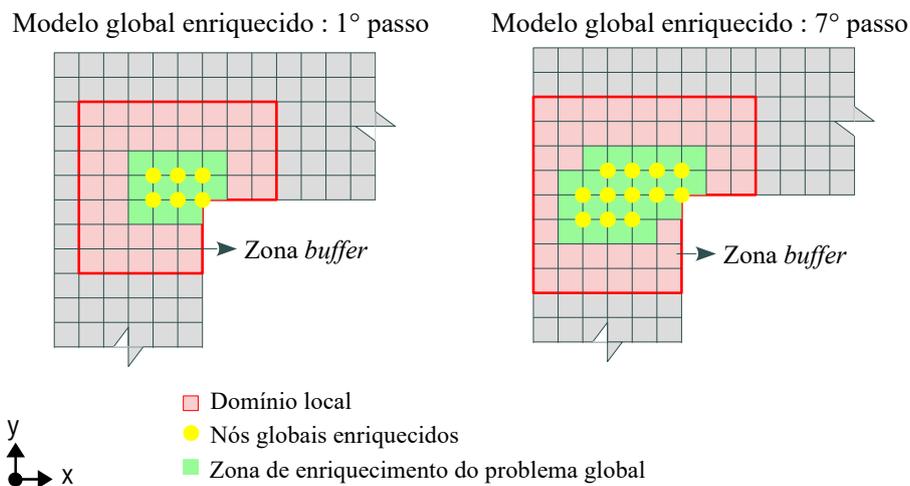


Figura 4.17: Problema global enriquecido no 1° e no 7° passo de propagação com zona *buffer* de duas camadas de elementos.

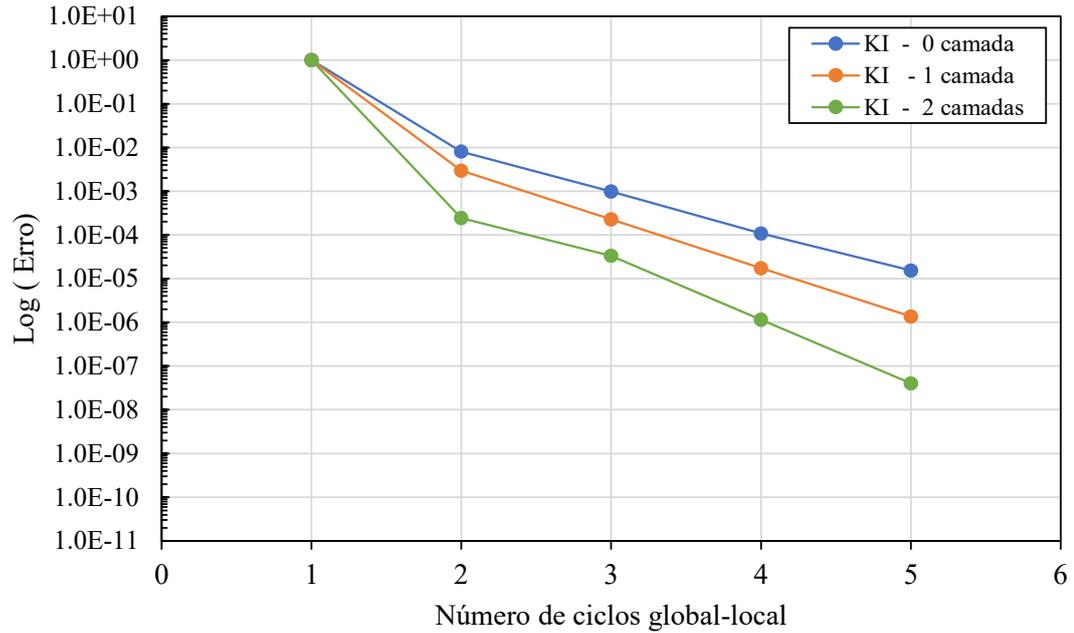


Figura 4.18: Comportamento da convergência de K_I em função do número de ciclos global-local e tamanho da zona buffer no 7º passo de propagação.

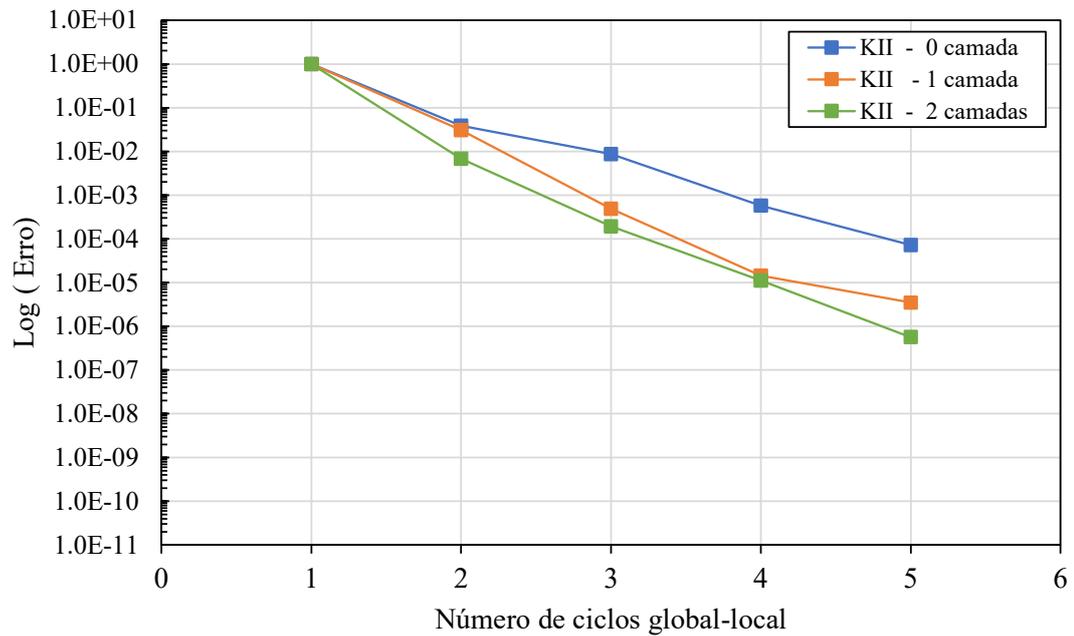


Figura 4.19: Comportamento da convergência de K_{II} em função do número de ciclos global-local e tamanho da zona buffer no 7º passo de propagação.

Considerar zona *buffer* com mais camadas de elementos seria uma possibilidade, mas não será realizado aqui devido a limitação da aplicação de mais de duas camadas de zona *buffer* no modelo da escala local das estratégias de acoplamento: adotar três camadas de zona *buffer*, por exemplo, faz parte dos limites do modelo local coincidir com os limites do modelo da mesoescala no 7º passo de propagação nas análises MEF G^{gl} -GLI.

Assim, a solução de referência utilizada para validar a implementação das estratégias $MEFG^{gl}$ -GLI corresponde a solução do modelo numérico equivalente considerando cinco ciclos de iteração global-local e duas camadas de zona *buffer*. A Fig. 4.20 apresenta os sete passos de propagação da trinca, representadas no modelo da escala local, obtidos pela solução de referência no modelo equivalente.

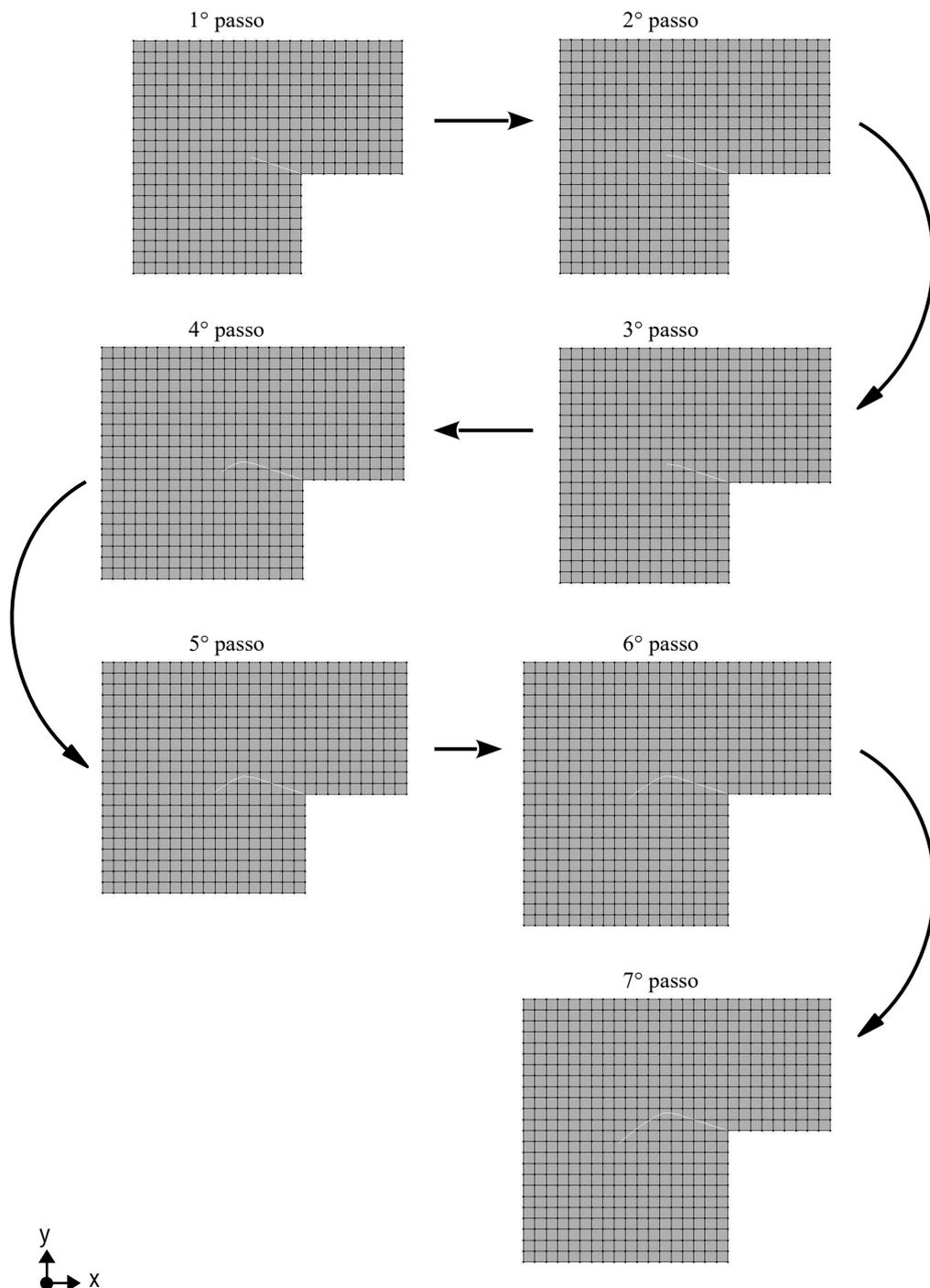


Figura 4.20: Passos de propagação da trinca obtidos pela solução de referência no modelo equivalente.

A partir de agora, a influência do número de ciclos global-local e tamanho da zona *buffer* serão avaliados considerando convergência da solução das estratégias de acoplamento.

4.2.2 Influência do número de ciclos global-local

No contexto das estratégias MFG^{gl}-GLI, Li *et al.* (2021) e Li *et al.* (2022) indicam que um único ciclo de iteração global-local ($n_{GL} = 1$) é suficiente para tratar os efeitos de condições de contorno de baixa precisão aplicadas ao modelo local. Porém, as iterações MFG^{gl} podem alterar a rigidez do modelo da mesoescala, e, dessa forma, afetar a taxa de convergência do GLI, já que, como mencionado na Seção 2.2, a convergência do GLI é determinada pela diferença de rigidez entre o domínio do modelo local (neste caso, mesoescala) e a correspondência deste domínio no modelo global.

Silveira Filho (2023), no tratamento de problemas elásticos lineares, concluiu que o incremento do número de ciclos global-local não melhorou, de maneira significativa, a qualidade dos resultados, tão pouco influenciou na convergência do GLI, mas onerou o procedimento de solução. Ainda assim, há interesse em investigar a influência do número de iterações global-local sobre a solução das estratégias MFG^{gl}-GLI para problemas da MFLE.

Assim, os resultados obtidos pelos procedimentos de solução monolítica (MFG^{gl}-GLI^{M2}) e *staggered* (MFG^{gl}-GLISt) do MFG^{gl}-GLI considerando um, dois, três, quatro e cinco ciclos global-local, e domínio local mínimo, ou seja, ausência de zona *buffer*, são avaliados e comparados com resultados obtidos pela solução de referência no modelo equivalente.

4.2.2.1 Fatores de intensidade de tensão, K_I e K_{II}

Os resultados em termos dos fatores de intensidade de tensão obtidos, a cada passo de propagação da trinca, pela solução de referência do modelo numérico equivalente e pelas estratégias MFG^{gl}-GLI^{M2} e MFG^{gl}-GLISt são tabulados, respectivamente, pelas Tabs. 4.8 e 4.9. O *Erro* se refere ao cálculo do erro relativo percentual entre os valores de K_I e K_{II} obtidos no último passo de propagação das estratégias MFG^{gl}-GLI em relação aos valores obtidos pela solução de referência também no último passo de propagação da trinca. O cálculo do erro em relação ao último passo foi definido como o cenário mais adverso para as comparações, por acumular o erro de aproximação das etapas anteriores de propagação da trinca. Os índices superiores dos fatores de intensidade de tensão, K_I e K_{II} , indicam, nas Tabs. 4.8 e 4.9, os resultados obtidos pela solução de referência, e o n_{GL} utilizado, quando obtidos pelas estratégias MFG^{gl}-GLI.

Verifica-se, a partir das Tabs. 4.8 e 4.9, que, assim como observado em Li *et al.* (2022), os resultados obtidos pelas estratégias MFG^{gl}-GLI^{M2} e MFG^{gl}-GLISt apresentam níveis de precisão quase idênticos entre si em termos de fatores de intensidade

de tensão. Observa-se ainda, que os maiores erros relativos obtidos no último passo de propagação estão associados à determinação de K_{II} , e são de, aproximadamente, 1, 80%.

Em relação a influência do número de ciclos global-local, observa-se que não há melhora significativa na precisão dos resultados em termos de K_I e de K_{II} com o aumento do número de ciclos global-local pelas estratégias de acoplamento. Verifica-se, na verdade, que os resultados apresentados nas Tabs. 4.8 e 4.9, referentes às estratégias MEFG^{gl}-GLI, no 1° e 7° passo de propagação, convergem para os mesmos valores de fatores de intensidade de tensão obtidos pelo modelo numérico equivalente também considerando domínio local mínimo, Tabs. 4.2 e 4.3. Percebe-se, contudo, que, embora as estratégias MEFG^{gl}-GLI não sejam capazes de melhorar a aproximação da solução de referência mantendo o domínio local mínimo e aumentando o número de ciclos global-local, elas conseguem, com um único ciclo global-local, produzir resultados suficientemente próximos dos resultados obtidos pelo modelo numérico equivalente via MEFG^{gl} quando neste adota-se $n_{GL} = 5$. Tal verificação é justificada pela repetição das simulações MEFG^{gl} ao longo do processo de solução das estratégias de acoplamento: ao invés de se ter n_{GL} ciclos de iteração global-local total, como ocorre na solução do modelo numérico equivalente, tem-se na verdade $n_{GL} \times n_i$ ciclos de iteração global-local, sendo n_i o número de iterações monolíticas ou *staggered* necessárias para atingir a convergência da estratégia.

A capacidade das estratégias MEFG^{gl}-GLI de aproximarem seus resultados, com um menor número de ciclos global-local, dos resultados obtidos pelo modelo numérico equivalente via MEFG^{gl}, torna as estratégias de acoplamento computacionalmente mais competitivas. Já que, como será visto na Seção 4.2.2.3, devido às múltiplas iterações do procedimento de solução, mesmo com um menor número de ciclos global-local, as estratégias MEFG^{gl}-GLI costumam requerer maior tempo computacional do que os procedimentos de solução convencionais.

Nesse sentido, com o objetivo de verificar o comportamento da convergência dos resultados em termos dos fatores de intensidade de tensão a partir da possível redução do número de iterações das estratégias de acoplamento, resolveu-se o problema da Fig. 4.13 utilizando a técnica de relaxação dinâmica. A técnica de relaxação é aplicada aqui apenas com o objetivo de acelerar o procedimento de solução, e a escolha pela relaxação dinâmica é para eliminar a necessidade de ser definir *a priori* um fator de relaxação ótimo.

As Tabs. 4.10 e 4.11 sintetizam os resultados em termos dos fatores de intensidade de tensão obtidos, a cada passo de propagação da trinca, pela solução de referência do modelo numérico equivalente, e pelas estratégias MEFG^{gl}-GLI^{M2} e MEFG^{gl}-GLISt adotando a técnica de relaxação dinâmica. Verifica-se assim, que a utilização da técnica de relaxação dinâmica não prejudica de forma significativa a aproximação dos resultados obtidos pelas estratégias de acoplamento em relação a solução de referência. Os erros relativos associados a determinação de K_I e K_{II} , no último passo de propagação de trinca, se mantiveram próximos de 0, 70% e 1, 80%, respectivamente. Esses resultados são obtidos

com um menor número de iterações das estratégias $\text{MEFG}^{gl}\text{-GLI}$, como será apresentado na Seção 4.2.2.3.

Assim, na determinação dos fatores de intensidade de tensão, a adoção das estratégias de acoplamento considerando $n_{GL} = 1$ e a técnica de relaxação dinâmica apresenta-se como uma alternativa mais interessante, tendo em vista o ganho em eficiência computacional, devido ao menor número de ciclos global-local e a aceleração da convergência, sem prejudicar a aproximação dos resultados. Tendo em vista as conclusões verificadas aqui, no contexto da aplicação das estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$, as análises que seguem serão realizadas com a utilização da técnica de relaxação dinâmica.

4.2.2.2 Deslocamento na direção y , u_y

Será verificada agora a influência do número de ciclos global-local sobre os resultados, em termos da componente de deslocamento y (u_y), na extremidade de aplicação da carga do pórtico apresentado na Fig. 4.13, onde é esperado deslocamento vertical mais significativo.

Assim, os deslocamentos verticais nos nós do modelo numérico equivalente na extremidade de aplicação da força são comparados com os deslocamentos verticais do nós correspondentes no modelo global das estratégias $\text{MEFG}^{gl}\text{-GLI}$. Como o modelo global do modelo numérico equivalente possui mesma discretização do modelo global das estratégias de acoplamento, há correspondência entre os nós na extremidade.

A partir da variação do número de ciclos global-local das estratégias $\text{MEFG}^{gl}\text{-GLI}$, verifica-se que não há alteração nos resultados em termos da componente de deslocamento na direção y , até a sexta casa decimal, com o aumento do número de ciclos. Tanto na estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$ quanto na estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$, os deslocamentos verticais obtidos por $n_{GL} = 1$ ou $n_{GL} = 5$ produziram o mesmo deslocamento. Além disso, os resultados obtidos pelas duas estratégias foram idênticos, o que corrobora as conclusões da Seção 4.2.2.1.

A Tab. 4.12, apresenta os resultados em termos da componente de deslocamento na direção y na extremidade direita do pórtico da Fig. 4.13, obtidos pela solução de referência e pelas estratégias $\text{MEFG}^{gl}\text{-GLI}$. É apresentado também o erro percentual relativo da solução das estratégias de acoplamento em relação a solução de referência do modelo numérico equivalente. Ainda, neste caso, a coluna $\text{MEFG}^{gl}\text{-GLI}^{M2,St}$ se refere aos resultados das estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$, uma vez que são iguais. A Fig. 4.21 apresenta os resultados do modelo numérico equivalente em termos da componente de deslocamento vertical.

Tabela 4.8: Resultados em termos dos fatores de intensidade de tensão obtidos pela solução de referência e pela estratégia MEF gl -GLI M2 .

Passos	K_I^{Ref}	K_{II}^{Ref}	$K_I^{(1)}$	$K_{II}^{(1)}$	$K_I^{(2)}$	$K_{II}^{(2)}$	$K_I^{(3)}$	$K_{II}^{(3)}$	$K_I^{(4)}$	$K_{II}^{(4)}$	$K_I^{(5)}$	$K_{II}^{(5)}$
1	4065,0876	-381,5177	4056,6910	-383,2944	4056,6897	-383,2947	4056,6898	-383,2947	4056,6898	-383,2947	4056,6898	-383,2947
2	3680,3579	-1120,5313	3684,0398	-1120,5746	3684,0353	-1120,5760	3684,0359	-1120,5758	3684,0359	-1120,5758	3684,0359	-1120,5758
3	2785,4180	-289,4418	2769,6207	-280,8601	2769,6211	-280,8581	2769,6209	-280,8583	2769,6210	-280,8583	2769,6210	-280,8583
4	2923,4358	-71,8582	2912,6373	-77,8944	2912,6383	-77,8948	2912,6376	-77,8948	2912,6378	-77,8948	2912,6378	-77,8948
5	2928,3682	-89,5748	2923,3733	-95,5433	2923,3759	-95,5425	2923,3746	-95,5427	2923,3749	-95,5427	2923,3748	-95,5426
6	2508,4537	-12,9947	2531,1589	-13,4509	2531,1625	-13,4510	2531,1608	-13,4512	2531,1612	-13,4511	2531,1611	-13,4512
7	2468,7755	-203,9109	2485,6760	-200,3050	2485,6767	-200,3060	2485,6763	-200,3053	2485,6764	-200,3055	2485,6764	-200,3054
<i>Erro (%)</i> :	-	-	0,6846	1,7683	0,6845	1,7679	0,6846	1,7682	0,6846	1,7681	0,6846	1,7682

Tabela 4.9: Resultados em termos dos fatores de intensidade de tensão obtidos pela solução de referência e pela estratégia MEF gl -GLI St .

Passos	K_I^{Ref}	K_{II}^{Ref}	$K_I^{(1)}$	$K_{II}^{(1)}$	$K_I^{(2)}$	$K_{II}^{(2)}$	$K_I^{(3)}$	$K_{II}^{(3)}$	$K_I^{(4)}$	$K_{II}^{(4)}$	$K_I^{(5)}$	$K_{II}^{(5)}$
1	4065,0876	-381,5177	4056,7019	-383,2928	4056,6925	-383,2962	4056,6999	-383,2940	4056,6994	-383,2942	4056,6994	-383,2942
2	3680,3579	-1120,5313	3684,0494	-1120,5738	3684,0575	-1120,5698	3684,0414	-1120,5701	3684,0368	-1120,5707	3684,0371	-1120,5707
3	2785,4180	-289,4418	2769,6263	-280,8662	2769,6336	-280,8585	2769,6335	-280,8604	2769,6357	-280,8581	2769,6352	-280,8584
4	2923,4358	-71,8582	2912,6513	-77,8840	2912,6593	-77,8972	2912,6595	-77,89843	2912,6628	-77,8951	2912,6619	-77,8951
5	2928,3682	-89,5748	2923,3924	-95,5496	2923,3977	-95,5410	2923,4046	-95,5403	2923,3985	-95,5443	2923,3998	-95,5434
6	2508,4537	-12,9947	2531,1847	-13,4542	2531,1885	-13,4492	2531,1944	-13,4506	2531,1921	-13,4472	2531,1916	-13,4484
7	2468,7755	-203,9109	2485,6871	-200,3060	2485,6861	-200,3110	2485,6903	-200,3139	2485,6919	-200,3130	2485,6917	-200,3121
<i>Erro (%)</i> :	-	-	0,6850	1,7679	0,6850	1,7654	0,6851	1,7640	0,6852	1,7644	0,6852	1,7649

Tabela 4.10: Resultados em termos dos fatores de intensidade de tensão obtidos pela solução de referência e pela estratégia MEFG^{gl}-GLI^{M2} adotando relaxação dinâmica.

Passos	K_I^{Ref}	K_{II}^{Ref}	$K_I^{(1)}$	$K_{II}^{(1)}$	$K_I^{(2)}$	$K_{II}^{(2)}$	$K_I^{(3)}$	$K_{II}^{(3)}$	$K_I^{(4)}$	$K_{II}^{(4)}$	$K_I^{(5)}$	$K_{II}^{(5)}$
1	4065,0876	-381,5177	4056,7182	-383,3049	4056,7103	-383,2931	4056,7132	-383,2886	4056,7130	-383,2890	4056,7130	-383,2890
2	3680,3579	-1120,5313	3684,0897	-1120,5918	3684,0863	-1120,5794	3684,0829	-1120,5767	3684,0831	-1120,5769	3684,0832	-1120,5771
3	2785,4180	-289,4418	2769,6439	-280,8648	2769,6359	-280,8815	22769,6427	-280,8861	2769,6434	-280,8858	2769,6434	-280,8856
4	2923,4358	-71,8582	2912,6715	-77,8999	2912,6706	-77,8926	2912,6672	-77,8893	2912,6678	-77,8902	2912,6676	-77,8901
5	2928,3682	-89,5748	2923,4164	-95,5482	2923,4213	-95,5630	2923,4151	-95,5607	2923,4163	-95,5613	2923,4161	-95,5610
6	2508,4537	-12,9947	2531,2092	-13,4478	2531,2214	-13,4679	2531,2128	-13,4610	2531,2146	-13,4625	2531,2142	-13,4622
7	2468,7755	-203,9109	2485,7159	-200,3373	2485,7372	-200,3566	2485,7246	-200,3473	2485,7274	-200,3495	2485,7267	-200,3489
<i>Erro (%)</i> :	-	-	0,6862	1,7525	0,6870	1,7431	0,6865	1,7476	0,6867	1,7465	0,6866	1,7468

Tabela 4.11: Resultados em termos dos fatores de intensidade de tensão obtidos pela solução de referência e pela estratégia MEFG^{gl}-GLISt adotando relaxação dinâmica.

Passos	K_I^{Ref}	K_{II}^{Ref}	$K_I^{(1)}$	$K_{II}^{(1)}$	$K_I^{(2)}$	$K_{II}^{(2)}$	$K_I^{(3)}$	$K_{II}^{(3)}$	$K_I^{(4)}$	$K_{II}^{(4)}$	$K_I^{(5)}$	$K_{II}^{(5)}$
1	4065,0876	-381,5177	4056,6932	-383,3026	4056,7078	-383,3024	4056,6996	-383,3028	4056,6990	-383,3029	4056,6991	-383,3029
2	3680,3579	-1120,5313	3684,0428	-1120,5943	3684,0896	-1120,5801	3684,0597	-1120,5849	3684,0550	-1120,5854	3684,0554	-1120,5855
3	2785,4180	-289,4418	2769,6449	-280,8561	2769,6446	-280,8694	22769,6481	-280,8616	2769,6503	-280,8595	2769,6499	-280,8596
4	2923,4358	-71,8582	2912,6703	-77,8945	2912,6782	-77,8961	2912,6790	-77,8983	2912,6824	-77,8991	2912,6815	-77,8992
5	2928,3682	-89,5748	2923,4163	-95,5523	2923,4272	-95,5505	2923,4326	-95,5480	2923,4209	-95,5527	2923,4260	-95,5504
6	2508,4537	-12,9947	2531,2104	-13,4703	2531,2130	-13,4640	2531,2166	-13,4652	2531,2163	-13,4600	2531,2157	-13,4637
7	2468,7755	-203,9109	2485,7210	-200,3260	2485,7190	-200,3326	2485,7230	-200,3345	2485,7241	-200,3356	2485,7242	-200,3330
<i>Erro (%)</i> :	-	-	0,6864	1,7581	0,6863	1,7548	0,6865	1,7539	0,6865	1,7534	0,6865	1,7546

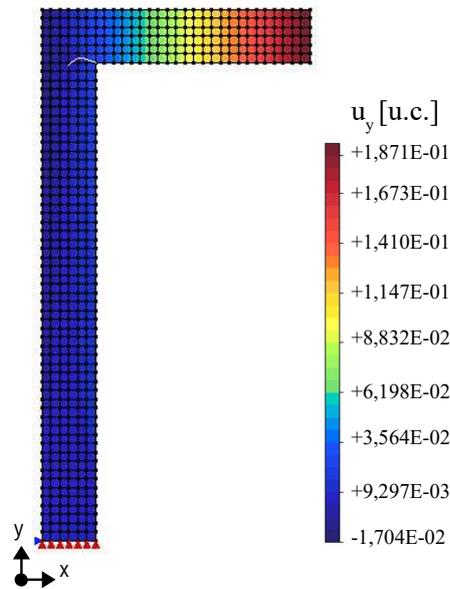


Figura 4.21: Resultados da solução de referência do modelo numérico equivalente em termos de u_y .

Tabela 4.12: Comparação dos resultados em termos de u_y ao longo da extremidade direita do pórtico, obtidos pela solução de referência do modelo numérico equivalente e pelas estratégias MEFG^{gl}-GLI adotando relaxação dinâmica.

Posição em y	Componente de deslocamento na direção y		Erro (%)
	Solução de referência	MEFG ^{gl} -GLI ^{M2,St}	
270	$1,8702 \times 10^{-1}$	$1,8700 \times 10^{-1}$	0,0107
275	$1,8703 \times 10^{-1}$	$1,8701 \times 10^{-1}$	0,0107
280	$1,8704 \times 10^{-1}$	$1,8702 \times 10^{-1}$	0,0107
285	$1,8711 \times 10^{-1}$	$1,8708 \times 10^{-1}$	0,0160
290	$1,8704 \times 10^{-1}$	$1,8702 \times 10^{-1}$	0,0107
295	$1,8703 \times 10^{-1}$	$1,8701 \times 10^{-1}$	0,0107
300	$1,8702 \times 10^{-1}$	$1,8700 \times 10^{-1}$	0,0107

Os dados apresentados na Tab. 4.12 revelam que os resultados obtidos pela solução de referência, em termos da componente de deslocamento na direção y , são ligeiramente superiores aos obtidos pelas estratégias de acoplamento. É um comportamento muito sutil, já que a superioridade está nas casas decimais. O maior erro verificado é de apenas 0,0160%, e está relacionado ao nó central da extremidade direita do pórtico, ou seja, ao nó onde o carregamento está sendo aplicado e que possui maior deslocamento vertical.

De toda forma, verifica-se a conformidade das soluções obtidas pelas estratégias MEFG^{gl}-GLI^{M2} e MEFG^{gl}-GLISt em relação aos resultados da solução de referência do modelo

numérico equivalente. Além disso, confirma-se a conclusão obtida na Seção anterior, da capacidade das estratégias de acoplamento produzirem bons resultados com o menor número de ciclos global-local e adotando a técnica de relaxação dinâmica.

4.2.2.3 Desempenho computacional

Essa Seção tem por objetivo verificar a influência do número de ciclos global-local sobre a convergência das estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$, além de comparar a eficiência computacional entre as estratégias de acoplamento, em função do tempo computacional, com e sem adoção da técnica de relaxação dinâmica. A verificação da eficiência computacional das estratégias $\text{MEFG}^{gl}\text{-GLI}$ em relação a solução via MEFG^{gl} convencional também é abordada aqui.

As Tabs. 4.13 e 4.14 apresentam a influência do número de ciclos global-local sobre a convergência e a eficiência computacional da estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$ sem e com relaxação dinâmica, respectivamente. Nas tabelas, n_{GL} corresponde ao número de ciclos global-local, enquanto $n_{i_{gl}}$, $n_{i_{gl}}$ e $n_{i_{tot}}$ referem-se, respectivamente, ao número de iterações GLI, global-local e totais realizadas até a convergência. Como na estratégia que implementa o algoritmo monolítico as iterações GLI e as simulações MEFG^{gl} ocorrem a cada iteração da estratégia, o número de iterações global-local é dado pelo produto entre o número de ciclos global-local e o número de iterações GLI ($n_{GL} \times n_{i_{gl}}$). Já o número de iterações totais é a soma das iterações GLI com as iterações global-local. O tempo computacional é dado por uma comparação relativa, e corresponde à razão entre o tempo computacional gasto para a respectiva solução e o tempo de execução da solução de referência.

Verifica-se inicialmente, a partir da Tab. 4.13, que o número de ciclos global-local não influencia na convergência da estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$, isto é, independente do número de ciclos global-local adotado a estratégia converge com 49 iterações. Nota-se, porém, que, como esperado, o aumento do número de ciclos global-local onera o procedimento de solução, aumentando o número de iterações totais do processo, e por consequência, aumentando também o tempo computacional.

Observa-se, ainda a partir da Tab. 4.14, que a utilização da técnica de relaxação dinâmica na estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$ provoca uma redução do número de iterações GLI, acelerando a convergência da solução. Verifica-se ainda, nesse caso, que o número de ciclos global-local também influencia na convergência da estratégia. Com um único ciclo global-local a convergência ocorre com 42 iterações GLI. O aumento do número de ciclos global-local até $n_{GL} = 3$ provoca uma redução do número de iterações GLI necessárias para a convergência. Com mais do que três ciclos global-local, porém, não são observadas alterações no número de iterações.

Tabela 4.13: Comparação da eficiência computacional da estratégia MCFG^{gl}-GLI^{M2} sem relaxação dinâmica.

n_{GL}	n_{gli}	n_{igl}	n_{tot}	tempo
1	49	49	98	1,91
2	49	98	147	3,15
3	49	147	196	4,32
4	49	196	245	5,57
5	49	245	294	6,85

Tabela 4.14: Comparação da eficiência computacional da estratégia MCFG^{gl}-GLI^{M2} com relaxação dinâmica.

n_{GL}	n_{gli}	n_{igl}	n_{tot}	tempo
1	42	42	84	1,69
2	39	78	117	2,50
3	38	114	152	3,38
4	38	152	190	4,36
5	38	190	228	5,72

As Tabs. 4.15 e 4.16 apresentam a influência do número de ciclos global-local sobre a convergência e a eficiência computacional da estratégia MCFG^{gl}-GLISt sem e com relaxação dinâmica, respectivamente. Nas tabelas, além das colunas presentes nas Tabs. 4.13 e 4.14, tem-se a coluna do número de iterações *staggered*, identificada por n_{is} . Assim, como no algoritmo de solução *staggered* as simulações MCFG^{gl} ocorrem apenas quando ocorre uma iteração *staggered*, o número de iterações global-local é dado pelo produto entre o número de ciclos global-local e o número de iterações *staggered* ($n_{GL} \times n_{is}$). O número de iterações totais permanece sendo a soma das iterações GLI com as iterações global-local, o tempo computacional ainda é o tempo relativo, e corresponde à razão entre o tempo computacional gasto para a respectiva solução e o tempo de execução da solução de referência.

Verifica-se, a partir da Tab. 4.15, que diferentemente do observado anteriormente na estratégia MCFG^{gl}-GLI^{M2}, o número de ciclos global-local influencia na convergência da estratégia MCFG^{gl}-GLISt sem relaxação. De maneira geral, o que se observa é que o aumento do número de ciclos global-local provoca a redução do número de iterações *staggered* e de iterações GLI. É importante lembrar aqui, que no algoritmo de solução *staggered* as simulações MCFG^{gl} não ocorrem a cada iteração da estratégia, como ocorre no algoritmo de solução monolítica. Tal fato justifica a maior sensibilidade da convergência

Tabela 4.15: Comparação da eficiência computacional da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ sem relaxação dinâmica.

n_{GL}	n_{i_s}	$n_{i_{gli}}$	$n_{i_{gl}}$	$n_{i_{tot}}$	tempo
1	35	86	35	121	2,24
2	31	79	62	141	2,77
3	28	70	84	154	3,18
4	30	73	120	193	4,17
5	30	72	150	222	5,11

Tabela 4.16: Comparação da eficiência computacional da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ com relaxação dinâmica.

n_{GL}	n_{i_s}	$n_{i_{gli}}$	$n_{i_{gl}}$	$n_{i_{tot}}$	tempo
1	35	68	35	103	1,93
2	31	64	62	126	2,54
3	28	55	84	139	2,93
4	30	58	120	178	3,88
5	29	57	145	202	4,69

da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ em relação ao número de ciclos global-local. O aumento do número de ciclos global-local melhora a qualidade da solução da mesoescala, acelerando, por consequência, a convergência do algoritmo GLI. Contudo, a redução do número de iterações da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ não é suficiente para compensar o aumento do tempo computacional devido ao maior número de ciclos global-local.

Ainda, a partir da Tab. 4.16, observa-se que a utilização da técnica de relaxação dinâmica na estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ provoca uma redução do número de iterações GLI, mantendo, exceto para $n_{GL} = 5$, o mesmo número de iterações *staggered* das soluções da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ sem considerar relaxação. Tal comportamento era esperado, já que as técnicas de relaxação atuam apenas sobre a convergência do algoritmo GLI. De modo geral, nota-se também que o aumento do número de ciclos global-local provoca uma redução do número de iterações GLI. Há, contudo, um comportamento anômalo a partir de $n_{GL} = 3$: quando $n_{GL} = 4$ e $n_{GL} = 5$ o número de iterações GLI, embora se mantenham próximos, são maiores do que quando $n_{GL} = 3$. Um comportamento semelhante da convergência da estratégia a partir de $n_{GL} = 3$ também foi observado na análise da estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$ com relaxação, e está possivelmente relacionado à estabilização numérica do processo de solução.

A Fig. 4.22 compara a eficiência computacional das estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$, sem e com a adoção da técnica de relaxação dinâmica, em função do

número de ciclos global-local.

Observa-se assim, a partir da Fig. 4.22, que ao comparar as estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$ sem relaxação, adotando um único ciclo global-local a estratégia que implementa o algoritmo de solução *staggered* requer maior tempo computacional relativo do que a estratégia que implementa o algoritmo monolítico. Porém, com o aumento do número de ciclos global-local a estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ se torna computacionalmente mais eficiente. Esse comportamento já havia sido observado nas investigações da Seção 4.1, e é mais uma vez verificado aqui.

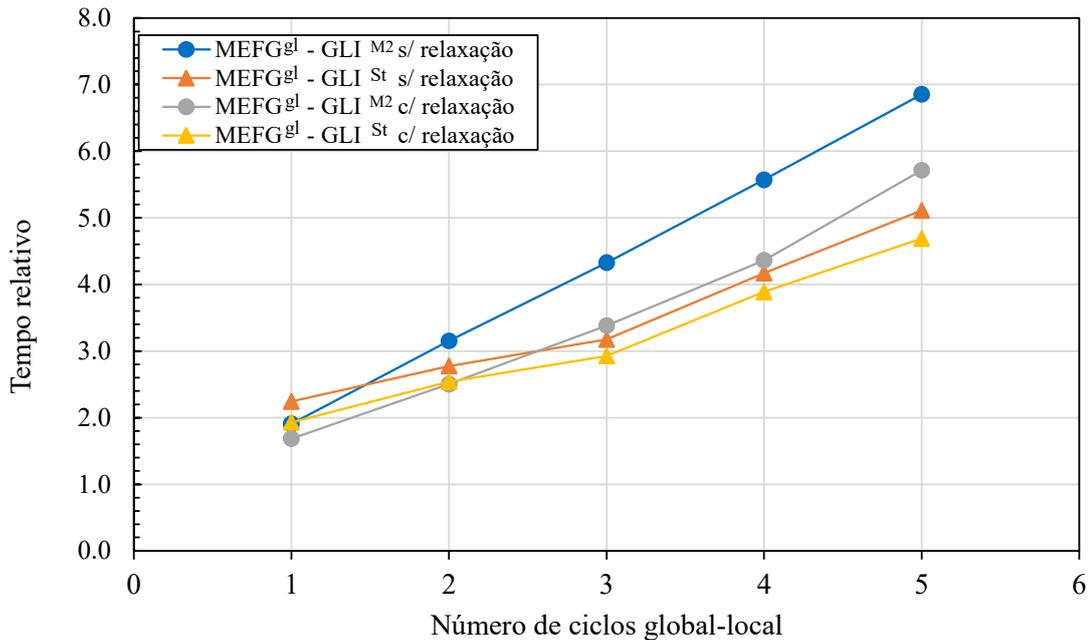


Figura 4.22: Comparação da eficiência computacional das estratégias $\text{MEFG}^{gl}\text{-GLI}^{St}$ e $\text{MEFG}^{gl}\text{-GLI}^{M2}$, sem e com relaxação dinâmica, em função do número de ciclos global-local.

De acordo com Li *et al.* (2022), a maior eficiência computacional da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ em relação a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$ está associada ao menor número de simulações MEFG^{gl} necessárias para se atingir a convergência. Contudo, neste caso, embora se tenha um menor número de iterações global-local, com $n_{GL} = 1$, o número de iterações totais (soma das iterações GLI com as iterações global-local) da estratégia que implementa o algoritmo *staggered* é superior ao número de iterações totais da estratégia que implementa o algoritmo monolítico, resultando no maior tempo computacional da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$. O aumento do número de ciclos global-local reduz o número das iterações *staggered* e GLI da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$, provocando a redução do número de iterações totais em relação as iterações totais da estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$. Além disso, o aumento do número de ciclos global-local onera as simulações MEFG^{gl} , que ocorrem a cada iteração da estratégia que implementa o algoritmo monolítico. Sendo assim, o aumento do número de ciclos global-local tem maior impacto sobre o custo computacional da estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$ do que da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$.

Comportamento semelhante é verificado ao se comparar as estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$ com relaxação. A diferença é que nesse caso, a estratégia que implementa o algoritmo *staggered* só se torna computacionalmente mais eficiente do que a estratégia que implementa o algoritmo monolítico quando $n_{GL} = 3$. A justificativa é a mesma descrita anteriormente, e está relacionada ao maior número de iterações totais necessárias pela estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ em relação a $\text{MEFG}^{gl}\text{-GLI}^{M2}$, quando $n_{GL} = 1$ e quando $n_{GL} = 2$. Verifica-se, contudo, que a melhora na eficiência computacional da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ com o aumento do número de ciclos global-local, adotando relaxação dinâmica, é menos significativa do que a observada pela estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$. O fato das técnicas de relaxação não atuarem na redução do número de iterações *staggered*, que são as iterações relacionadas com as simulações MEFG^{gl} , no contexto da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$, faz com que o ganho em eficiência computacional da estratégia que implementa o algoritmo *staggered*, com adoção da técnica de relaxação, ser menos significativo e menos influenciado pelo número de ciclos global-local.

Verifica-se, por fim, que a solução da estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$ adotando a técnica de relaxação dinâmica e $n_{GL} = 1$ é a que apresenta melhor custo benefício, já que, como verificado nas seções anteriores, produz resultados adequados com menor custo computacional. No entanto, mesmo a abordagem mais eficiente das estratégias de acoplamento requer cerca de 70% mais tempo computacional do que a solução via MEFG^{gl} convencional. Tal fato é justificado pelo grande número de iterações necessárias para se atingir a convergência da solução, ao mesmo tempo que as iterações das estratégias $\text{MEFG}^{gl}\text{-GLI}$ fazem com que a solução convirja com menor número de ciclos global-local, elas também podem tornar as estratégias computacionalmente menos eficientes.

4.2.3 Influência do tamanho da zona *buffer*

Como mencionado na Seção 2.3.4, assim como os ciclos global-local, a zona *buffer* é uma estratégia utilizada na solução do MEFG^{gl} para melhorar a precisão da solução do método, minimizando os impactos de erros advindos da transferência da solução global para o problema local como condição de contorno.

Essa Seção tem por objetivo investigar a influência do tamanho da zona *buffer* no contexto das estratégias de acoplamento $\text{MEFG}^{gl}\text{-GLI}$. Assim, os resultados obtidos pelos procedimentos de solução monolítica ($\text{MEFG}^{gl}\text{-GLI}^{M2}$) e *staggered* ($\text{MEFG}^{gl}\text{-GLI}^{St}$) do $\text{MEFG}^{gl}\text{-GLI}$ considerando domínio local mínimo, uma e duas camadas de elementos de zona *buffer*, são avaliados e comparados com os resultados obtidos pela solução de referência.

A partir das conclusões das seções anteriores, as investigações da influência da zona *buffer* sobre o comportamento das estratégias de acoplamento são realizadas assumindo $n_{GL} = 1$ e relaxação dinâmica. Já a solução de referência é adotada como a solução do

modelo numérico equivalente considerando zona *buffer* com duas camadas de elementos, e $n_{GL} = 5$.

4.2.3.1 Fatores de intensidade de tensão, K_I e K_{II}

Os resultados em termos dos fatores de intensidade de tensão obtidos, a cada passo de propagação da trinca, pela solução de referência do modelo numérico equivalente e pelas estratégias $MEFG^{gl}$ - GLI^{M2} e $MEFG^{gl}$ - GLI^{St} são tabulados, respectivamente, pelas Tabs. 4.17 e 4.18. O *Erro* se refere ao cálculo do erro relativo percentual entre os valores de K_I e K_{II} obtidos no último passo de propagação das estratégias $MEFG^{gl}$ - GLI em relação aos valores obtidos pela solução de referência também no último passo de propagação da trinca. O cálculo do erro em relação ao último passo foi definido como o cenário mais adverso para as comparações, por acumular o erro de aproximação das etapas anteriores de propagação da trinca. Os índices superiores dos fatores de intensidade de tensão, K_I e K_{II} , indicam, nas Tabs. 4.17 e 4.18, os resultados obtidos pela solução de referência, e o número de camadas de elementos de zona *buffer* utilizadas, quando obtidas com as estratégias $MEFG^{gl}$ - GLI .

A solução obtida pelas estratégias de acoplamento, considerando ausência de zona *buffer* e um único ciclo global-local, já foi apresentada anteriormente, e, de acordo com as Tabs. 4.17 e 4.18, é a que pior aproxima a solução de referência em termos dos fatores de intensidade de tensão. Com o aumento da zona *buffer*, as soluções das estratégias $MEFG^{gl}$ - GLI melhoram a aproximação em relação a solução de referência quando a zona *buffer* possui duas camadas de elementos. Assim, mesmo considerando $n_{GL} = 1$ e técnica de relaxação dinâmica, quando são consideradas duas camadas de elementos de zona *buffer* os resultados em termos dos fatores de intensidade de tensão obtidos pelas estratégias de acoplamento apresentam boa aproximação da solução de referência, de modo que o erro relativo associado a determinações de K_I e K_{II} no último passo de propagação é da ordem de 10^{-2} .

Verifica-se que os resultados obtidos pelas estratégias $MEFG^{gl}$ - GLI^{M2} e $MEFG^{gl}$ - GLI^{St} apresentam níveis de precisão quase idênticos em termos de fatores de intensidade de tensão. Esse comportamento já havia sido observado nas simulações anteriores e se mantêm aqui. A conformidade dos resultados em termos dos fatores de intensidade de tensão, entre as estratégias $MEFG^{gl}$ - GLI e a solução de referência, também se reflete no comportamento de propagação da trinca, que seguiu o mesmo caminho apresentado na Fig. 4.20.

Tabela 4.17: Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente e pela estratégia MEFG^{gl}-GLI^{M2} adotando relaxação dinâmica e diferentes camadas de zona *buffer*.

Passos	K_I^{Ref}	K_{II}^{Ref}	$K_I^{(0)}$	$K_{II}^{(0)}$	$K_I^{(1)}$	$K_{II}^{(1)}$	$K_I^{(2)}$	$K_{II}^{(2)}$
1	4065,0876	-381,5177	4056,7182	-383,3049	4065,3653	-381,6989	4065,2251	-381,5959
2	3680,3579	-1120,5313	3684,0897	-1120,5917	3681,0149	-1121,1286	3680,5492	-1120,7010
3	2785,4180	-289,4418	2769,6439	-280,8648	2783,1494	-284,6965	2785,5750	-289,3210
4	2923,4358	-71,8582	2912,6715	-77,8999	2919,2541	-73,1457	2923,5598	-71,9343
5	2928,3682	-89,5748	2923,4164	-95,5482	2923,7356	-89,2031	2928,4989	-89,5296
6	2508,4537	-12,9947	2531,2092	-13,4478	2503,5346	-14,1587	2508,5710	-13,0466
7	2468,7755	-203,9109	2485,7159	-200,3373	2467,1684	-208,5759	2468,9093	-203,8946
<i>Erro (%)</i> :	-	-	0,6862	1,7525	0,0651	2,2878	0,0054	0,0080

Tabela 4.18: Resultados em termos dos fatores de intensidade de tensão obtidos pelo modelo numérico equivalente e pela estratégia MEFG^{gl}-GLISt adotando relaxação dinâmica e diferentes camadas de zona *buffer*.

Passos	K_I^{Ref}	K_{II}^{Ref}	$K_I^{(0)}$	$K_{II}^{(0)}$	$K_I^{(1)}$	$K_{II}^{(1)}$	$K_I^{(2)}$	$K_{II}^{(2)}$
1	4065,0876	-381,5177	4056,6932	-383,3026	4065,3604	-381,7049	4065,2172	-381,6036
2	3680,3579	-1120,5313	3684,0428	-1120,5943	3681,0153	-1121,1360	3680,5465	-1120,7182
3	2785,4180	-289,4418	2769,6449	-280,8561	2783,1684	-284,6865	2785,5799	-289,3101
4	2923,4358	-71,8582	2912,6703	-77,8945	2919,2579	-73,1572	2923,5647	-71,9349
5	2928,3682	-89,5748	2923,4163	-95,5523	2923,7423	-89,1981	2928,5056	-89,5298
6	2508,4537	-12,9947	2531,2104	-13,4703	2503,5444	-14,1683	2508,5774	-13,0493
7	2468,7755	-203,9109	2485,7210	-200,3260	2467,1779	-208,5773	2468,9128	-203,9020
<i>Erro (%)</i> :	-	-	0,6864	1,7581	0,0647	2,2884	0,0056	0,0044

Tabela 4.19: Comparação dos resultados em termos de u_y ao longo da extremidade direita do pórtico, obtidos pela solução de referência do modelo numérico equivalente e pelas estratégias $\text{MEFG}^{gl}\text{-GLI}$ adotando relaxação dinâmica e diferentes camadas de zona *buffer*.

Posição em y	Solução de referência	$\text{MEFG}^{gl}\text{-GLI}^{M2,St}$		
		0 camada	1 camada	2 camadas
270	$1,8702 \times 10^{-1}$	$1,8700 \times 10^{-1}$	$1,8704 \times 10^{-1}$	$1,8704 \times 10^{-1}$
275	$1,8703 \times 10^{-1}$	$1,8701 \times 10^{-1}$	$1,8704 \times 10^{-1}$	$1,8704 \times 10^{-1}$
280	$1,8704 \times 10^{-1}$	$1,8702 \times 10^{-1}$	$1,8706 \times 10^{-1}$	$1,8706 \times 10^{-1}$
285	$1,8711 \times 10^{-1}$	$1,8708 \times 10^{-1}$	$1,8712 \times 10^{-1}$	$1,8712 \times 10^{-1}$
290	$1,8704 \times 10^{-1}$	$1,8702 \times 10^{-1}$	$1,8706 \times 10^{-1}$	$1,8706 \times 10^{-1}$
295	$1,8703 \times 10^{-1}$	$1,8701 \times 10^{-1}$	$1,8704 \times 10^{-1}$	$1,8704 \times 10^{-1}$
300	$1,8702 \times 10^{-1}$	$1,8700 \times 10^{-1}$	$1,8704 \times 10^{-1}$	$1,8704 \times 10^{-1}$

4.2.3.2 Deslocamento na direção y , u_y

Será verificada agora a influência do tamanho da zona *buffer* sobre os resultados, em termos da componente de deslocamento y (u_y), na extremidade de aplicação da carga do pórtico apresentado na Fig. 4.13. Assim, são avaliados os deslocamentos verticais nodais dos nós localizados na extremidade direita do pórtico.

De forma semelhante ao que foi observado na investigação da influência do número de ciclos global-local sobre as estratégias $\text{MEFG}^{gl}\text{-GLI}$, foi verificado aqui que as estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$ produzem os mesmos resultados em termos da componente de deslocamento na direção y , até a sexta casa decimal.

A Tab. 4.19 a seguir, apresenta os resultados em termos da componente de deslocamento na direção y na extremidade direita do pórtico da Fig. 4.13, obtidos pela solução de referência do modelo numérico equivalente e pelas estratégias $\text{MEFG}^{gl}\text{-GLI}$, em função do tamanho da zona *buffer*. Ainda, neste caso, a coluna $\text{MEFG}^{gl}\text{-GLI}^{M2,St}$ refere-se aos resultados das estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$, que são coincidentes, com as casas decimais utilizadas.

Verifica-se, a partir da Tab. 4.19, que a solução obtida, em termos da componente de deslocamento na direção y , considerando uma camada de zona *buffer* é ligeiramente superior a obtida pela solução na qual foi admitido domínio local mínimo. Porém, o acréscimo de camadas de zona *buffer*, de uma para duas camadas, não produz alterações nos resultados em termos de deslocamento vertical. Nota-se assim, que, semelhante ao que ocorre com o número de ciclos global-local, o tamanho da zona *buffer* interfere mais localmente, no cálculo dos fatores de intensidade de tensão (K_I e K_{II}), do que nos resultados em termos da componente de deslocamento vertical.

Nota-se também, de acordo com a Tab. 4.19, que os deslocamentos verticais obtidos

pelas estratégias de acoplamento, com uma e com duas camadas de elementos, são ligeiramente superiores aos obtidos pela solução de referência. Contudo, o deslocamento vertical do nó central da extremidade direita do pórtico, que possui maior deslocamento vertical, obtido pelas soluções que consideram uma e duas camadas de elementos de zona *buffer*, está mais próximo da solução de referência, do que o deslocamento obtido pela solução com ausência de zona *buffer*. Este é o ponto de aplicação da força, e, portanto, é a partir deste resultado que se pode calcular a energia de deformação, parâmetro adequado para se avaliar o comportamento geral da solução.

De toda forma, verifica-se a conformidade das soluções obtidas pelas estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$ em relação aos resultados da solução de referência do modelo numérico equivalente. Assim, como nesse caso, a influência do tamanho da zona *buffer* nos resultados em termos da componente de deslocamento vertical é muito sutil, a escolha de quantas camadas de zona *buffer* adotar pode estar atrelada ao custo computacional, que será avaliado na Seção seguinte.

4.2.3.3 Desempenho computacional

Essa Seção tem por objetivo verificar a influência do tamanho da zona *buffer* sobre a convergência das estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$, além de comparar a eficiência computacional entre as estratégias de acoplamento, e com relação a solução via MEFG^{gl} convencional.

As Tabs. 4.20 e 4.21 apresentam a influência do tamanho da zona *buffer* sobre a convergência e a eficiência computacional das estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$, respectivamente. Na Tab. 4.20 a primeira coluna corresponde ao número de camadas de elementos que compõem a zona *buffer*, enquanto $n_{i_{gli}}$, $n_{i_{gl}}$ e $n_{i_{tot}}$ referem-se, respectivamente, ao número de iterações GLI, global-local e totais realizadas até a convergência. Como na estratégia que implementa o algoritmo monolítico, as iterações GLI e as simulações MEFG^{gl} ocorrem a cada iteração da estratégia, o número de iterações global-local é dado pelo produto entre o número de ciclos global-local e o número de iterações GLI ($n_{GL} \times n_{i_{gli}}$) e, no caso, $n_{GL} = 1$. Já o número de iterações totais é a soma das iterações GLI com as iterações global-local. O tempo computacional é dado por uma comparação relativa, e corresponde à razão entre o tempo computacional gasto para a respectiva solução e o tempo de execução da solução de referência.

Observa-se, a partir da Tab. 4.20, que o tamanho da zona *buffer* influencia na convergência da estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$. De modo que, o aumento do número de camadas de elementos que compõem a zona *buffer* provoca a diminuição do número de iterações da estratégia, reduzindo o tempo de execução do procedimento de solução, e aumentando a eficiência computacional. Verifica-se, na verdade, uma relação entre as iterações GLI e as soluções global-local, a exemplo do que foi discutido na Seção 4.2.2.3. Aqui apenas uma iteração global-local está sendo aplicada e o erro advindo da imposição das condições de

Tabela 4.20: Comparação da eficiência computacional da estratégia $MEFG^{gl}$ -GLI M2 adotando relaxação dinâmica e diferentes camadas de zona *buffer*.

zona <i>buffer</i>	$n_{i_{gli}}$	$n_{i_{gl}}$	$n_{i_{tot}}$	tempo
0 camada	42	42	84	1,69
1 camada	38	38	76	1,57
2 camadas	34	34	68	1,51

Tabela 4.21: Comparação da eficiência computacional da estratégia $MEFG^{gl}$ -GLI St adotando relaxação dinâmica e diferentes camadas de zona *buffer*.

zona <i>buffer</i>	n_{i_s}	$n_{i_{gli}}$	$n_{i_{gl}}$	$n_{i_{tot}}$	tempo
0 camada	35	68	35	103	1,93
1 camada	31	59	31	90	1,78
2 camadas	26	52	26	78	1,62

contorno no modelo local é minimizado com o aumento da zona *buffer*. Quanto menor a zona *buffer*, maior será o impacto deste erro que acaba tendo que ser corrigido, na presença de apenas um ciclo global-local, pelas iterações GLI. A cada nova iteração GLI, uma solução mais aprimorada da mesoescala, devido ao enriquecimento global-local da iteração anterior, é utilizada para impor os deslocamentos ao problema local de uma nova solução global-local. Dessa forma, o aumento da zona *buffer* reduz o erro da imposição das condições de contorno no modelo local, diminuindo o número de iterações GLI. No caso do exemplo numérico utilizado, esta redução está compensando e superando o custo computacional oriundo da solução do modelo da escala local, reduzindo, assim, o tempo computacional total da simulação.

A Tab. 4.21 apresenta a influência do tamanho da zona *buffer* sobre a convergência e a eficiência computacional da estratégia $MEFG^{gl}$ -GLI St . Na Tab. 4.21, além das colunas presentes na Tab. 4.20, tem-se a coluna do número de iterações *staggered*, identificada por n_{i_s} . Assim, como no algoritmo de solução *staggered* as simulações $MEFG^{gl}$ ocorrem apenas quando ocorre uma iteração *staggered*, o número de iterações global-local é dado pelo produto entre o número de ciclos global-local e o número de iterações *staggered* ($n_{GL} \times n_{i_s}$), no caso, $n_{GL} = 1$. O número de iterações totais permanece sendo a soma das iterações GLI com as iterações global-local, o tempo computacional ainda é o tempo relativo, e corresponde à razão entre o tempo computacional gasto para a respectiva solução e o tempo de execução da solução de referência.

De acordo com a Tab. 4.21, verifica-se que, assim como acontece com a estratégia $MEFG^{gl}$ -GLI M2 , o tamanho da zona *buffer* influencia na convergência da estratégia $MEFG^{gl}$ -GLI St . O aumento do número de camadas de elementos que compõem a zona *buffer* provoca a redução do número de iterações *staggered* e de iterações GLI, reduzindo o número

de iterações totais e melhorando a eficiência computacional da estratégia.

De toda forma, verifica-se que, para o problema em estudo (Fig. 4.13), a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$ com um único ciclo global-local, duas camadas de elementos de zona *buffer*, e relaxação dinâmica, é a estratégia de acoplamento computacionalmente mais eficiente. Ainda que a estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ exija um menor número de simulações MEFG^{gl} , computacionalmente mais onerosas, o maior número de iterações totais, tornam a estratégia que implementa o algoritmo *staggered* menos eficiente do que a estratégia que implementa o algoritmo monolítico.

Contudo, é válido registrar aqui, que, ainda que não tenha sido possível observar de forma explícita nos resultados apresentados nas Tabs. 4.20 e 4.21, o ônus do aumento do modelo da escala local, devido ao aumento do tamanho da zona *buffer*, pode ser menos significativo na estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ do que na estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$. Isso se dá pois, na estratégia que implementa o algoritmo *staggered*, a solução do problema local ocorre apenas nas iterações *staggered*, enquanto que na estratégia que implementa o algoritmo monolítico a solução do modelo da escala local ocorre em cada iteração da estratégia. Isso pode ser observado se for calculado a redução percentual no tempo da simulação de 0 para 2 camadas. Para o $\text{MEFG}^{gl}\text{-GLI}^{M2}$ a redução foi de 11% e para o $\text{MEFG}^{gl}\text{-GLI}^{St}$ ela foi de 16%.

Tal observação não se restringe apenas ao aumento do tamanho do modelo da escala local, mas a qualquer intervenção que onere o seu procedimento de solução. Para demonstrar essa observação, o problema apresentado na Fig. 4.13 foi solucionado a partir do refinamento da malha do modelo da escala local, que passou a ser definido pela divisão das arestas dos elementos do modelo da mesoescala por nove, resultando em elementos com dimensão $l = 0,5556$. Para que o refinamento da malha não altere a região do modelo da escala local com funções de trinca, foi definido raio de enriquecimento de 1,18, dado pela metade do comprimento da diagonal dos elementos do modelo da escala local do modelo equivalente.

A Tab. 4.22 sintetiza as informações referentes aos números de iterações e tempo computacional obtidos pelas estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$ considerando o novo modelo da escala local, duas camadas de zona *buffer*, e técnica de relaxação dinâmica. As identificações das colunas da Tab. 4.22 seguem sendo as mesmas utilizadas nas Tabs. 4.20 e 4.21. O tempo computacional também permanece sendo a razão entre o tempo computacional gasto para a respectiva solução e o tempo de execução da solução de referência originalmente definida.

Verifica-se, a partir da Tab. 4.22 que, como esperado, o refinamento do modelo da escala local onera significativamente o processo de solução de ambas as estratégias de acoplamento. Contudo, observa-se que a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$ requer cerca de 3% mais tempo computacional do que a estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$. Ou seja, sob estas condições, a estratégia que implementa o algoritmo *staggered* é computacionalmente mais eficiente

Tabela 4.22: Comparação da eficiência computacional das estratégias $\text{MEFG}^{gl}\text{-GLI}^{M2}$ e $\text{MEFG}^{gl}\text{-GLI}^{St}$, adotando novo modelo da escala local, duas camadas de zona *buffer*, e relaxação dinâmica.

Solução	n_{i_s}	$n_{i_{gli}}$	$n_{i_{gl}}$	$n_{i_{tot}}$	tempo
$\text{MEFG}^{gl}\text{-GLI}^{M2}$	-	30	30	60	7,47
$\text{MEFG}^{gl}\text{-GLI}^{St}$	24	41	24	65	7,26

que o algoritmo monolítico. Assim, a estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ continua a demandar maior número de iterações totais do que a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$, o fato, contudo, da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$ resolver o problema local, que está computacionalmente mais caro, apenas nas iterações *staggered*, melhora o seu desempenho computacional em relação a estratégia $\text{MEFG}^{gl}\text{-GLI}^{M2}$, que soluciona o modelo da escala local a cada iteração.

Ao comparar os resultados apresentados na Tab. 4.22 com aqueles apresentados ao considerar duas camadas de zona *buffer* nas Tabs. 4.20 e 4.21, percebe-se ainda que o refinamento da malha do modelo da escala local provoca a redução do número de iterações necessárias para a convergência das estratégias de acoplamento, sobretudo no número de iterações GLI da estratégia $\text{MEFG}^{gl}\text{-GLI}^{St}$. De modo geral, a redução no número de iterações está relacionado a melhora na precisão da solução do MEFG^{gl} , devido ao refinamento do modelo da escala local. Porém, é importante destacar que, independente do refinamento do modelo da escala local, serão sempre adicionados dois graus de liberdade a cada nó enriquecido do modelo da mesoescala. Assim, a maior ou menor rigidez do modelo da mesoescala, que irá influenciar na convergência do GLI, é determinada pela solução advinda do modelo da escala local. Desse modo, a redução do número de iterações GLI com o aumento do refinamento do modelo local não deve ser entendido como um comportamento geral do processo de solução das estratégias de acoplamento.

CAPÍTULO 5

CONSIDERAÇÕES FINAIS

5.1 Considerações deste Trabalho

Neste trabalho foi realizada a ampliação da estratégia de acoplamento não intrusivo entre o *software* comercial *Abaqus* e o ambiente computacional INSANE, através da estratégia MEFG^{gl} -GLI, proposta inicialmente por Li *et al.* (2021), e seu desdobramento em Li *et al.* (2022). Assim, a estratégia MEFG^{gl} -GLI que já havia sido implementada por Silveira Filho (2023), em sua abordagem monolítica, para simulações de problemas com comportamento linear elástico, foi expandida aqui dando ênfase ao tratamento de problemas da MFLE. Tal expansão permitiu a descrição e propagação de trinca na escala local, e implementou o algoritmo de solução *staggered* da estratégia MEFG^{gl} -GLI.

Na implementação proposta neste trabalho, o modelo global é solucionado pelo *software* comercial *Abaqus* via MEF, enquanto que o enriquecimento do modelo da mesoescala a partir da solução do modelo da escala local é realizado pelo MEFG^{gl} implementado no ambiente INSANE. O algoritmo que gerencia o acoplamento, juntamente com as técnicas de relaxação, foram desenvolvidos dentro do próprio ambiente INSANE para acoplar o modelo da escala global e o modelo da mesoescala. Assim, diferentemente de Silveira Filho (2023), que desenvolveu uma aplicação independente para acoplar o *software* *Abaqus* com o ambiente INSANE, aqui denominada MEFG^{gl} -GLI^{M1}, neste trabalho, a não intrusividade da implementação é somente em relação ao *software* comercial. A comunicação do ambiente INSANE e o *Abaqus* é realizada por intermédio de *scripts* desenvolvidos em linguagem de programação Python (Python, 2021), que são executados dentro do ambiente INSANE através do projeto Jython (Jython, 2022). Numa estratégia, denominada MEFG^{gl} -GLI^{M2}, observou-se a convergência com menos iterações do que na MEFG^{gl} -GLI^{M1}.

A validação da implementação proposta foi realizada a partir de simulações numéricas que investigaram parâmetros relacionados ao MEFG^{gl} sobre o comportamento da estratégia iterativa do acoplamento não intrusivo. Os resultados obtidos pelas estratégias de acoplamento foram comparados com soluções de referência, construídos por modelos de elementos finitos com malha mais refinada ou de discretização equivalente à malha do modelo da escala local.

Inicialmente, com o objetivo de validar a implementação do algoritmo de solução *staggered* da estratégia MEFG^{gl}-GLI (MEFG^{gl}-GLISt), o primeiro exemplo numérico apresentado consistiu na solução de uma chapa tracionada com um orifício quadrado sob EPT. Os resultados obtidos pela estratégia MEFG^{gl}-GLISt em termos da componente de deslocamento na direção y foram coincidentes com os resultados obtidos por Silveira Filho (2023), ao implementar o algoritmo de solução monolítica (MEFG^{gl}-GLI^{M1}), e apresentaram conformidade com os resultados obtidos posicionando-se entre a solução equivalente (Modelo A) e a solução de referência (Modelo B). Os resultados obtidos pela estratégia MEFG^{gl}-GLISt em termos da componente de tensão normal na direção x também apresentaram boa aproximação nos pontos de análise, mostrando-se mais próximos dos resultados do Modelo A. Em relação ao desempenho computacional, a estratégia MEFG^{gl}-GLISt foi menos eficiente do que o procedimento de solução monolítica para o MEFG^{gl}-GLI implementado no ambiente INSANE (MEFG^{gl}-GLI^{M2}), ao adotar um único ciclo de iteração global-local. Com o aumento do custo computacional associado às simulações MEFG^{gl} (aumento do número de ciclos global-local, aumento do número de pontos integração, etc.), a estratégia MEFG^{gl}-GLISt torna-se computacionalmente mais eficiente do que a estratégia MEFG^{gl}-GLI^{M2}.

Os algoritmos de solução da estratégia MEFG^{gl}-GLI também foram aplicados na resolução de um pórtico em “L” com a presença e a propagação de uma trinca. Para este caso, foram investigados a influência do número de ciclos global-local, do tamanho da zona *buffer*, e da utilização da técnica de relaxação dinâmica sobre os resultados obtidos pelas estratégias de acoplamento em termos dos fatores de intensidade de tensão, do deslocamento na direção y e do desempenho computacional.

Na avaliação da influência do número de ciclos global-local (n_{GL}), observou-se que não há melhora significativa na precisão dos resultados em termos dos fatores de intensidade de tensão, e da componente de deslocamento na direção y , com o aumento do número de ciclos global-local pelas estratégias de acoplamento. Verificou-se, contudo, que devido a repetição das simulações MEFG^{gl} ao longo do processo de solução, as estratégias de acoplamento conseguem, com um único ciclo global-local, produzir resultados suficientemente próximos daqueles obtidos via MEFG^{gl} convencional a partir de um maior número de ciclos. Isso ocorre pois a solução global-local mesmo com um único ciclo é repetida ao longo das iterações do acoplamento GLI.

Na avaliação da influência do tamanho da zona *buffer*, verificou-se que, o aumento do número de camadas de zona *buffer* melhorou a aproximação das soluções das estratégias de MEFG^{gl}-GLI em relação a solução de referência em termos dos fatores de intensidade de tensão e da componente de deslocamento na direção y .

Ainda, observou-se que os resultados obtidos pelas estratégias monolítica e *staggered* produziram, até a sexta casa decimal, os mesmos resultados em termos da componente de deslocamento na direção y , e apresentaram níveis de precisão quase idênticos entre si em

termos de fatores de intensidade de tensão, assim como foi observado em Li *et al.* (2022).

Em relação ao desempenho computacional, verificou-se que, como esperado, tanto na estratégia $MEFG^{gl}\text{-GLI}^{M2}$ quanto na estratégia $MEFG^{gl}\text{-GLI}^{St}$, o aumento do número de ciclos global-local onerou o procedimento de solução. Porém, diferentemente do que ocorre na estratégia $MEFG^{gl}\text{-GLI}^{M2}$, onde o número de ciclos global-local não influencia na convergência da solução, na estratégia $MEFG^{gl}\text{-GLI}^{St}$ o aumento do número de ciclos global-local provoca a redução do número de iterações da estratégia. Constatou-se também que, como esperado, a utilização da técnica de relaxação dinâmica reduz o número de iterações GLI, em ambas as estratégias $MEFG^{gl}\text{-GLI}$. Ainda, a redução do número de iterações não prejudicou de forma significativa a acurácia dos resultados, justificando a utilização da técnica nas análises numéricas realizadas.

Foi observado também, que o aumento do tamanho da zona *buffer* influencia na convergência das estratégias $MEFG^{gl}\text{-GLI}$. Tanto na estratégia $MEFG^{gl}\text{-GLI}^{M2}$ como na estratégia $MEFG^{gl}\text{-GLI}^{St}$, o aumento do número de camadas de elementos que compõem a zona *buffer* reduzem o número de iterações necessárias para se atingir a convergência da solução. Assim, ainda que o aumento do tamanho da zona *buffer* provoque o aumento do tamanho do modelo da escala local, onerando o seu processo de solução, foi verificado que a redução do número de iterações necessárias para a convergência da estratégia, pode compensar e superar o custo da solução do modelo da escala local, reduzindo o tempo computacional final.

Ao comparar o desempenho computacional da estratégia que implementa o algoritmo *staggered* com a estratégia que implementa o algoritmo monolítico, observou-se que, para o problema do pórtico em “L” com propagação de trinca, a estratégia $MEFG^{gl}\text{-GLI}^{M2}$ com um único ciclo global-local ($n_{GL} = 1$), duas camadas de zona *buffer*, e relaxação dinâmica é a estratégia de acoplamento computacionalmente mais eficiente. Contudo, também foi verificado que o aumento do custo da solução do modelo da escala local tem maior impacto sobre a estratégia $MEFG^{gl}\text{-GLI}^{M2}$ do que sobre a estratégia $MEFG^{gl}\text{-GLI}^{St}$, assim, na simulação em que foi realizado o refinamento da malha local e o aumento da nuvem de enriquecimento do modelo da escala local, a estratégia que implementa o algoritmo *staggered* foi mais eficiente do que a estratégia que implementa o algoritmo monolítico.

Portanto, assim como foi concluído na solução do exemplo numérico da chapa tracionada com o orifício, quanto maior o custo computacional associado às simulações $MEFG^{gl}$ (solução do modelo da escala local, enriquecimento do modelo da mesoescala e solução da mesoescala enriquecida) mais relevante pode ser a eficiência computacional da estratégia $MEFG^{gl}\text{-GLI}^{St}$ em relação a $MEFG^{gl}\text{-GLI}^{M2}$. Tal fato justifica a redução de 75% de tempo computacional total pela estratégia que implementa o algoritmo *staggered* em relação a estratégia que implementa o algoritmo monolítico quando em Li *et al.* (2022), são acoplados modelos sólidos e de casca 3D.

Finalmente, em relação ao desempenho computacional inferior das estratégias de acoplamento em relação à solução convencional, é importante pontuar que o objetivo principal deste trabalho não foi o de propor uma solução computacionalmente mais eficiente, mas sim de disponibilizar uma ferramenta capaz de integrar métodos numéricos avançados, implementados no ambiente INSANE, com a alta eficiência computacional do *software* comercial *Abaqus*. Sendo assim, o ganho em eficiência computacional irá de fato ocorrer quando o modelo global da estrutura for muito mais complexo, de modo que sua solução apenas no INSANE não seja eficiente o suficiente para competir com o *Abaqus*.

5.2 Sugestões para Trabalhos Futuros

Com base neste trabalho, podem ser sugeridos os seguintes tópicos como propostas para trabalhos futuros:

1. Aplicar as estratégias de acoplamento não intrusivo para a solução de problemas de maior complexidade da escala global, que de fato justifique a utilização do *software* comercial.
2. Adaptar a implementação que gerencia o acoplamento não intrusivo para solução de outras classes de problemas, que envolvam fadiga e não linearidade física ou geométrica, por exemplo.
3. Adaptar a implementação que gerencia o acoplamento para análises multidomínios, que permita a solução de mais de um problema local (e sua respectiva mesoescala) para um mesmo problema global. Neste caso, considerar utilizar computação paralela para as soluções locais.
4. Adaptar a implementação que gerencia o acoplamento para compatibilizar o campo de deslocamentos ao longo da interface entre a escala global e a mesoescala, quando no modelo da escala global for utilizado grau de aproximação maior do que um, e no modelo da mesoescala for utilizada aproximação equivalente, mas advinda do enriquecimento polinomial. Por exemplo, em um problema bidimensional, utilizar aproximação quadrática no modelo da escala global, resultaria na utilização de elementos finitos do tipo Q8, com três nós na interface por elemento e cada nó com dois graus de liberdade em cada direção, totalizando seis graus de liberdade. Já na mesoescala, utilizar elementos finitos do tipo Q4 enriquecidos por um monômio, resultando em dois nós na interface por elemento com três graus de liberdade em cada direção para cada nó, totalizando doze graus de liberdade;
5. Em complemento a sugestão anterior, adaptar a implementação que gerencia o acoplamento não intrusivo para permitir o uso de modelos de análise distintos, a exemplo de (Li *et al.*, 2022), que realizou o acoplamento multiescala de modelos sólidos e de casca 3D.

BIBLIOGRAFIA

- Abaqus (2014), ‘Version 6.14 documentation’. Dassault Systemes Simulia Corporation, Providence, RI, USA.
- 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* pp. 203–220.
- Alves, P. D. (2012), Estratégia global-local aplicada ao método dos elementos finitos generalizados, Master’s thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.
- Alves, P. D., Barros, F. B. e Pitangueira, R. L. S. (2013), ‘An object-oriented approach to the generalized finite element method’, *Advances in Engineering Software* vol. 59, 1–18.
- Barros, F. B. (2002), Métodos Sem Malha e Métodos dos Elementos Finitos Generalizados em Análise Não-Linear de Estruturas, PhD thesis, EESC - USP, São 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.
- Bettinotti, O., Allix, O. e Malherbe, B. (2014), ‘A coupling strategy for adaptive local refinement in space and time with a fixed global model in explicit dynamics’, *Computational Mechanics* vol. 53(4), 561–574.
- Birner, M. e Schweitzer, M. A. (2019), ‘Global-local enrichments in puma’, In ‘*Mesh-free Methods for Partial Differential Equations IX*’. Springer International Publishing, pp. 167–183.
- Botelho, G. G., Monteiro, H. A. S., Peixoto, R. G. e Pitangueira, R. L. S. (2018), ‘Object oriented framework for multiphysics analysis’, *XXXIX Ibero-Latin American Congress on Computational Methods in Engineering* .
- 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.

- Curreli, C., Puccio, F. D. e Mattei, L. (2018), ‘Application of the finite element submodeling technique in a single point contact and wear problem’, *International Journal for Numerical Methods in Engineering* vol. 116(10-11), 708–722.
- Duarte, C. A., Babuška, I. e Oden, J. O. (2000), ‘Generalized finite element methods for three-dimensional structural mechanics problems’, *Computers & Structures* vol. 77(2), 215–232.
- Duarte, C. A. 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.
- Duval, M., Passieux, J. C., Salaun, 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.
- Figueiredo, A. F., Castro, S. S., Pitangueira, R. L. S. e Saliba, S. S. (2020), ‘A machine learning-based constitutive model for nonlinear analysis via finite element method’, *XLI Ibero-Latin-American Congress on Computational Methods in Engineering*.
- Filmore, 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), 2.
- Fonseca, G. M. (2019), Propagação de trincas em meios elásticos lineares via método dos elementos finitos generalizados com estratégia global-local automatizada, Master’s thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.
- Fonseca, G. M. (2023), Estimadores de erro a *posteriori* e estratégias adaptativas para o Método dos Elementos Finitos Generalizados com enriquecimento Global-Local aplicado a problemas da Mecânica da Fratura, PhD thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.
- Fries, T. P. e Belytschko, T. (2010), ‘The extended/generalized finite element method: an overview of the method and its applications’, *International journal for numerical methods in engineering* vol. 84(3), 253–304.
- Gendre, L., Allix, O., Gosselet, P. e François, C. (2009), ‘Non-intrusive and exact global/local techniques for structural problems with local plasticity’, *Computational Mechanics* vol. 44, 233–245.
- Gori, L., Penna, S. S. e Pitangueira, R. L. S. (2018), ‘Smoothed point interpolation methods for the regularization of material instabilities in scalar damage models’, *International Journal for Numerical Methods in Engineering* pp. 1–27.

- Gori, L., Penna, S. S. e Pitangueira, R. L. S. (2019), ‘A computational framework for the constitutive modeling of nonlinear micropolar media’, *Journal of the Brazilian Society of Mechanical Sciences and Engineering* vol. 41, 1–21.
- 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.
- Guinard, S., Bouclier, R., Toniolli, M. e Passieux, J. C. (2018), ‘Multiscale analysis of complex aeronautical structures using robust non-intrusive coupling’, *Advanced Modeling and Simulation in Engineering Sciences* vol. 5(1), 1–27.
- 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, 47–62.
- Gupta, V., Pereira, J., Kim, D. J., Duarte, C. A. 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.
- Irons, B. M. e Tuck, R. C. (1969), ‘A version of the aitken accelerator for computer iteration’, *International Journal of Numerical Methods in Engineering* vol. 1, 275–277.
- Jython (2022), ‘Jython 2.7.3’.
URL: www.jython.org
- Kim, D. J., Pereira, J. P. e Duarte, C. A. (2010), ‘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.
- Li, H., Avecillas-Leon, J., Shauer, N. e Duarte, C. A. (2022), ‘A non-intrusive iterative generalized finite element method for multiscale coupling of 3-d solid and shell models’, *Computer Methods in Applied Mechanics and Engineering* vol. 402, 115408.
- Li, H., O’hara, P. e Duarte, C. A. (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.
- Liu, Y. J., Sun, Q. e Fan, X. L. (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.
- Malekan, M., Silva, L. L., Barros, F. B., Pitangueira, R. L. S. e Penna, S. S. (2018a), ‘Fracture analysis in plane structures with the two-scale g/xfem method’, *International Journal of Solids and Structures* vol. 155, 65–80.

- Malekan, M., Silva, L. L., Barros, F. B., Pitangueira, R. L. S. e Penna, S. S. (2018b), ‘Two-dimensional fracture modeling with the generalized/extended finite element method: An object oriented programming approach’, *Engineering Analysis with Boundary Elements* vol. 136, 186–203.
- Monteiro, H. A. S., Novelli, L., Fonseca, G. M., Pitangueira, R. L. S. e Barros, F. B. (2020), ‘A new approach for physically nonlinear analysis of continuum damage mechanics problems using the generalized/extended finite element method with global-local enrichment’, *Engineering Analysis With Boundary Elements* vol. 113, 277–295.
- Noor, A. K. (1986), ‘Global-local methodologies and their application to nonlinear analysis’, *Finite Elements in Analysis and Design* vol. 2(4), 333–346.
- Oden, J. T., Duarte, C. A. M. e Zienkiewicz, O. C. (1996), ‘A new cloud-based *hp* finite element method’, *Technical Report TICAM Report 96-55* .
- Oliveira, D. B., Penna, S. S. e Pitangueira, R. L. S. (2020), ‘Elastoplastic constitutive modeling for concrete: a theoretical and computational approach’, *Revista Ibracon de Estruturas e Materiais* vol. 13, 171–182.
- Peixoto, R. G., Ribeiro, G. O. e Pitangueira, R. L. S. (2018), ‘A boundary element method formulation for quasi brittle material fracture analysis using the continuum strong discontinuity approach’, *Engineering Fracture Mechanics* vol. 202, 47–74.
- Pereira, L. C. B. (2019), Implementação para o método global-local na plataforma insane, Master’s thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.
- Pinheiro, D. C. C., Barros, F. B. e Pitangueira, R. L. S. (2022), ‘Global-local analysis with element free galerkin method’, *Engineering Analysis with Boundary Elements* vol. 136, 186–203.
- Pitangueira, R. L. S. e Fonseca, F. T. (2007), ‘Insane: uma plataforma para computação científica’, *X Encontro de Modelagem Computacional* .
- 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’, *XXVII Latin American Congress on Computational Methods in Engineering* pp. 1–20.
- Plews, J., Duarte, C. A. 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.
- Python (2021), ‘Versão 3.10’. Python Software Foundation.
URL: www.python.org

- Silva, L. L. (2016), Sistema gráfico interativo para análise de nucleação e propagação de trinca, Master's thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.
- Silveira Filho, N. A. (2023), Implementação não intrusiva do método dos elementos finitos generalizados global-local, Master's thesis, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.
- Szabo, B. e Babuška, I. (1991), *Finite Element Analysis*, John Wiley & Sons, Inc.
- Whitcomb, J. D. (1991), 'Iterative global-local finite element analysis', *Computers & Structures* vol. 40(4), 1027–1031.
- Wolff, K. P., Pitangueira, R. L. S. e Penna, S. S. (2010), 'Um modelo propagação de fissuras no concreto baseado no método dos elementos finitos estendido', *Mecânica Computacional* vol. 29(102), 1013–10153.
- Xu, J. (1992), 'Iterative methods by space decomposition and subspace correction', *SIAM review* vol. 34(4), 581–613.

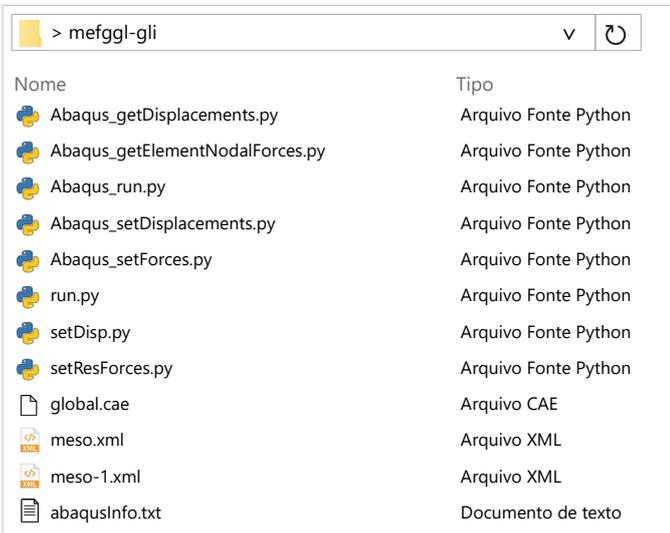
Apêndices

APÊNDICE A

REQUISITOS

A implementação desenvolvida neste trabalho requer apenas a instalação do *software* comercial Abaqus e da versão do ambiente INSANE que contém as classes que realizam o acoplamento. Não é necessário realizar a instalação de linguagens de programação adicionais. O código do projeto Jython, por exemplo, foi incorporada como uma dependência ao projeto do INSANE, através do arquivo `pom.xml` (*Project Object Model*) do pacote *Solution*. Dessa maneira, a compilação do código ocorre de forma automatizada.

Contudo, a execução da implementação requer que, além dos arquivos referentes aos modelos numéricos das escalas global, meso e local, os *scripts* Python, que realizam a comunicação do ambiente INSANE com o Abaqus, também estejam contidos na pasta de trabalho (Fig. A.1).



Nome	Tipo
Abaqus_getDisplacements.py	Arquivo Fonte Python
Abaqus_getElementNodalForces.py	Arquivo Fonte Python
Abaqus_run.py	Arquivo Fonte Python
Abaqus_setDisplacements.py	Arquivo Fonte Python
Abaqus_setForces.py	Arquivo Fonte Python
run.py	Arquivo Fonte Python
setDisp.py	Arquivo Fonte Python
setResForces.py	Arquivo Fonte Python
global.cae	Arquivo CAE
meso.xml	Arquivo XML
meso-1.xml	Arquivo XML
abaqusInfo.txt	Documento de texto

Figura A.1: Pasta de trabalho com os arquivos de entrada.

Os arquivos de extensão `.py` se referem aos *scripts* Python. O arquivo de extensão `.cae` corresponde ao modelo numérico da escala global, enquanto os arquivos de extensão `.xml` representam os modelos numéricos da escala meso e local. O arquivo de texto `.txt` contém informações sobre o projeto do modelo global. A identificação dos arquivos de entrada é realizada de forma automática, a partir do caminho do arquivo de entrada (`meso.xml`), e por isso não devem ter os nomes alterados.

A execução da implementação requer ainda especificações relacionadas ao projeto do modelo da escala global no Abaqus. Na construção deste modelo, deve ser definido um conjunto de nós (`Model > Assembly > Sets`) correspondente aos nós da interface entre o modelo global e o modelo da mesoescala. O nome do modelo criado, e o nome dado ao conjunto de nós deve ser informado no arquivo de texto `abaqusInfo.txt`, e será utilizado para indicar ao Abaqus quais nós do modelo global receberão informações advindas do modelo da mesoescala.

Outro ponto importante em relação ao projeto do Abaqus está relacionado às configurações da análise de processamento. Neste ponto é necessário criar e configurar o `Job` (`Analysis > Jobs`). Na configuração da precisão do `Job` deve-se adotar precisão *“full”*. A escolha da precisão influencia na alocação de memória durante o processamento do modelo, e, embora a precisão *“full”* ocupe mais memória do que a precisão *“single”*, que seria a outra opção possível, ela vai garantir maior precisão durante os cálculos e na saída de dados do processamento. Numa análise iterativa, que envolve tolerâncias e resíduos de ordem muito baixa, como é o caso da estratégia $MEFG^{gl}$ -GLI, a precisão *“full”* é a que irá garantir a acurácia da solução. O nome dado ao `Job` criado também deverá ser informado no arquivo de texto `abaqusInfo.txt`, pois será utilizado para acioná-lo através dos scripts Python.

O arquivo `abaqusInfo.txt`, mencionado anteriormente, é um arquivo de texto que contém informações específicas sobre o projeto do modelo global. Tais informações são lidas e armazenadas na classe `IGLGlobalLocal`, e são utilizadas internamente na execução dos *scripts* Python. Todos os arquivos apresentados na Fig. A.1 estão disponíveis em: <https://github.com/anaclarabueno/mefggl-gli.git>.

APÊNDICE B

DADOS DE ENTRADA

A entrada de dados da implementação ocorre a partir do arquivo de entrada `.xml` correspondente ao modelo numérico da mesoescala (`meso.xml`). Assim, além das informações sobre o modelo da mesoescala, esse arquivo deve conter os dados referentes à estratégia de acoplamento. A Fig. B.1 mostra os dados de entrada para definição da solução `IGLGlobalLocal`.

```

1 <Solution class="IGLGlobalLocal">
2   <IterativeGlobalLocal>
3     <Type>MONOLITHIC</Type>
4     <RelaxType>STATIC</RelaxType>
5     <RelaxCoefficient>1.00</RelaxCoefficient>
6     <IGLTolerance>1.0E-5</IGLTolerance>
7     <GlobalInterfaceElements>8 28 48 68 88 108 13 33 53 73 93 113</
      GlobalInterfaceElements>
8     <GlobalInterfaceNodes>9 30 51 72 93 114 135 13 34 55 76 97 118 139</
      GlobalInterfaceNodes>
9     <MesoInterfaceElements>E3 E11 E19 E27 E35 E43 E6 E14 E22 E30 E38 E46</
      MesoInterfaceElements>
10    <MesoInterfaceNodes>5 6 21 30 39 48 57 13 14 25 34 43 52 61</MesoInterfaceNodes>
11  </IterativeGlobalLocal>
12    <LocalPath>-1</LocalPath>
13    <GlobalSolverType>201</GlobalSolverType>
14    <LocalSolverType>201</LocalSolverType>
15    <GlobalLocalSolverType>201</GlobalLocalSolverType>
16    <GlobalLocalIterations>1</GlobalLocalIterations>
17 </Solution>

```

Figura B.1: Arquivo de entrada `meso.xml` para definição da solução `IGLGlobalLocal`.

A tag `<Type>` se refere ao tipo de procedimento de solução do `MEFGgl-GLI` que será utilizado, e pode ser `MONOLITHIC` ou `STAGGERED`. O elemento dessa tag será armazenado na forma de uma `String` na classe `IGLGlobalLocal`.

A tag `<RelaxType>` está relacionada ao tipo de técnica de relaxação que será utilizada, e pode ser `STATIC` ou `DYNAMIC`, para se referir a técnica de relaxação estática e dinâmica, respectivamente. Na tag seguinte, `<RelaxCoefficient>`, é informado o fator de relaxação. Na aplicação da técnica de relaxação estática o valor informado ao fator de relaxação será

utilizado para corrigir a solução em todas as iterações da estratégia. Para este caso, adotar fator de relaxação 1,0 corresponde a não aplicar correção à solução. Já na aplicação da técnica de relaxação dinâmica, o valor informado ao fator de relaxação será o valor utilizado nas duas primeiras iterações da estratégia, nas iterações seguintes o fator de relaxação será atualizado. Os elementos das tags `<RelaxType>` e `<RelaxCoefficient>` são armazenados na classe `IGLGlobalLocal` na forma de uma `String` e de um `double`, respectivamente.

A tag `<IGLTolerance>` se refere a tolerância do GLI, e é utilizado na verificação da convergência do algoritmo GLI ao longo da estratégia $MEFG^{gl}$ -GLI. Neste trabalho, foi adotado tolerâncias da ordem de 10^{-5} , em conformidade com os trabalhos de Li *et al.* (2021) e Silveira Filho (2023).

As tags `<GlobalInterfaceElements>` e `<GlobalInterfaceNodes>` se referem, respectivamente, aos elementos e aos nós da interface do modelo global. E as tags `<MesoInterfaceElements>` e `<MesoInterfaceNodes>` estão relacionadas aos elementos e aos nós da interface do modelo da mesoescala. A identificação dos nós e dos elementos dos modelos da escala global e meso, ao longo da interface, deve ser feita conforme apresentado na Fig. B.2. A sequência com que os elementos e os nós são inseridos no arquivo `.xml` deve ser a mesma para os dois modelos, já que é a partir dessa informação que a implementação identifica a correspondência entre os nós do modelo global e do modelo da mesoescala. Os elementos dessas tags são armazenados na classe `IGLGlobalLocal` na forma de `ArrayList<String>`.

As demais tags fazem parte da definição da solução `GlobalLocal` que é estendida para a definição da solução `IGLGlobalLocal` devido ao mecanismo de herança entre as classes.

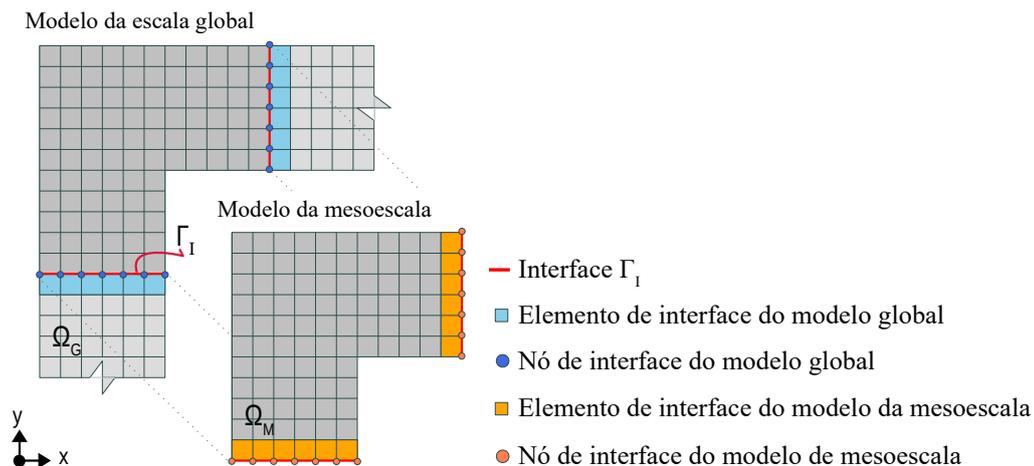


Figura B.2: Identificação dos nós e dos elementos de interface.

Em relação a construção do modelo numérico da mesoescala, os nós da interface do modelo da mesoescala devem ter os graus de liberdade restritos no arquivo de entrada, uma vez que, durante o procedimento de solução, estes nós receberão deslocamentos

prescritos advindos da solução do modelo global. Já o modelo numérico da escala local (`meso-1.xml`) tem sua existência condicionada pela abordagem do MEF gl utilizada. Na utilização da abordagem do MEF gl aplicada a problemas com propagação de trinca, por exemplo, o modelo local é gerado de forma automatizada, o que elimina a necessidade do arquivo de entrada de dados do modelo da escala local.