



Ministério da Educação

Universidade Federal da Integração Latino-Americana
Instituto Latino-Americano de Tecnologia, Infraestrutura e Território
Centro Interdisciplinar de Tecnologia e Infraestrutura

Engenharia Civil de Infraestrutura

ANÁLISE TÉRMO-MECÂNICA 2D DA BARRAGEM DE CONTRAFORTE DA ITAIPU PELO MÉTODO DOS ELEMENTOS FINITOS EM FORTRAN

EDIVALDO JOSÉ DA SILVA JUNIOR

Foz do Iguaçu, PR
Outubro de 2018



Ministério da Educação

Universidade Federal da Integração Latino-Americana
Instituto Latino-Americano de Tecnologia, Infraestrutura e Território
Centro Interdisciplinar de Tecnologia e Infraestrutura

Engenharia Civil de Infraestrutura

ANÁLISE TÉRMO-MECÂNICA 2D DA BARRAGEM DE CONTRAFORTE DA ITAIPU PELO MÉTODO DOS ELEMENTOS FINITOS EM FORTRAN

EDIVALDO JOSÉ DA SILVA JUNIOR

Plano de Trabalho apresentado à Banca Examinadora do
Curso de Engenharia Civil de Infraestrutura da UNILA,
como parte dos requisitos para obtenção do Grau de
Bacharel em Engenharia Civil.

Orientador: Profº. Drº. AREF KALILO LIMA KZAM

Foz do Iguaçu, PR
Outubro de 2018

AGRADECIMENTOS

Agradeço primeiramente aos meus pais, Edivaldo e Silvana, por terem se abdicado de seus interesses pessoais após o meu nascimento. Agradeço também a minha irmã mais nova, Sabrina. Juntos, compartilhamos inúmeros momentos difíceis, mas também, infinitos momentos felizes. Nada foi fácil. Nada foi gratuito. Todos nós sacrificamos vontades e necessidades para o bem da família. E embora hoje, cada um tenha ido para uma direção diferente, sempre teremos a forte conexão familiar, que vai além do sangue e do poder financeiro.

Ninguém é nada sozinho. Para alcançar todas as minhas conquistas tive muitos amigos, mestres e professores, que me ajudaram e acreditaram em mim. A todos vocês deixo os meus mais sinceros agradecimentos. E gostaria de citar alguns que fizeram a diferença e com certeza são responsáveis pelo caminho que segui: Os professores Jonatas e Edneia da Banda Marcial; Sensei George Osako, Sandro e Shinjiro Toyomura da Academia de Judô e o Engenheiro Dimilson da CHI.

Agradeço ao PTI-CEASB e a ITAIPU Binacional, pelos recursos tecnológicos e financeiros disponibilizados para a elaboração deste trabalho.

Agradeço também a minha noiva, Joara, que se tornou a minha parceira e melhor amiga na vida desde o primeiro momento que nos conhecemos.

Finalmente, agradeço ao meu orientador, Aref Kzam, que me ajudou a despertar o interesse em programação de métodos numéricos para resolução de problemas de engenharia.

SILVA JUNIOR, E. J. Análise **Termo-Mecânica 2D da Barragem de Contraforte da Itaipu Pelo Método dos Elementos Finitos em Fortran**. Trabalho de Conclusão de Curso (Graduação em Engenharia Civil de Infraestrutura) – Universidade Federal da Integração Latino-Americana, Foz do Iguaçu, 2018.

RESUMO

As barragens de concreto estão sujeitas a diversos tipos de carregamentos, provenientes do peso próprio, da pressão hidrostática do reservatório e da variação térmica sazonal. Diversos trabalhos destacam a importância de se avaliar os efeitos da variação de temperatura em relação ao desempenho e a durabilidade das estruturas. Dessa forma, a avaliação do campo de temperatura é essencial para se determinar tensões e deslocamentos de origem térmica. Através de um estudo de caso na Barragem da Central Hidrelétrica de Itaipu (CHI), é proposto o desenvolvimento de um código computacional em Fortran, com base no método dos elementos finitos, capaz de simular o comportamento termo-mecânico de um meio bidimensional submetido à transferência de calor por condução. As propriedades térmicas do concreto foram obtidas em relatórios de ensaio da barragem, disponível no Sistema de Arquivo Técnico (SAT) da CHI. Os dados dos instrumentos de monitoramento da barragem foram utilizados para se determinar as condições de contorno e validação. O acoplamento é realizado empregando-se o campo de temperatura obtido na análise térmica como condição inicial de carregamento na análise mecânica. A fim de complementar a etapa de validação realizou-se a mesma análise com o programa comercial ANSYS®, cuja eficiência é comprovadamente reconhecida na literatura técnica e o acoplamento é realizado de maneira intuitiva com as ferramentas atualmente disponíveis no *workbench*. As soluções das simulações foram analisadas e os resultados do código desenvolvido foram satisfatórios.

Palavras-chave: Itaipu, Barragem de Contrafortes; Transferência de calor; Termo-Mecânico; MEF; Fortran

SILVA JUNIOR, E. J. **2d Thermo-Mechanical Analysis Of Itaipu Buttress Dam Using The Finite Element Method In Fortran.** Trabalho de Conclusão de Curso (Graduação em Engenharia Civil de Infraestrutura) – Universidade Federal da Integração Latino-Americana, Foz do Iguaçu, 2018.

ABSTRACT

Concrete dams are subject to different types of loads: their own weight, hydrostatic reservoir pressure and seasonal thermal variation. Several studies emphasize the importance of evaluating the effects of temperature variation in relation to the performance and durability of the structures. Thus, the evaluation of the temperature field is essential to determine thermal stresses and displacements. Through a case study at the ITAIPU Hydroelectric Power Station (CHI), it is proposed the development of a computational code in FORTRAN, based on the Finite Element Method, capable of simulating the thermal behavior of a two dimensional domain subjected to conduction heat transfer. The thermal properties of the concrete were obtained in dam test reports, available from the CHI Technical File System (SAT). The dam monitoring instruments data were used to determine the boundary and validation conditions. Thermo-structural coupling is performed using the temperature field as the nodal contour condition via the initial thermal deformation. In order to validate the model, the ANSYS® commercial software was used. Finally, the solutions of the simulations were analyzed and the results of the developed code were satisfactory.

Keywords: Itaipu, buttress dam, Heat transfer, Thermo-Mechanical , FEA, Fortran

SUMÁRIO

AGRADECIMENTOS.....	iii
RESUMO.....	iv
ABSTRACT	v
INTRODUÇÃO.....	10
OBJETIVOS	12
OBJETIVO GERAL.....	12
OBJETIVOS ESPECÍFICOS.....	12
1. REVISÃO BIBLIOGRÁFICA	13
1.1. TRANSFERÊNCIA DE CALOR	15
1.1.1. CONDUÇÃO.....	15
1.1.2. CONVECÇÃO	17
1.1.3. RADIAÇÃO.....	18
1.2. TENSÕES TÉRMICAS	19
1.3. PROPRIEDADES DO CONCRETO.....	20
1.3.1. MÓDULO DE ELASTICIDADE (E).....	20
1.3.2. COEFICIENTE DE POISSON (ν).....	20
1.3.3. CONDUTIVIDADE TÉRMICA (K)	21
1.3.4. COEFICIENTE DE DILATAÇÃO TÉRMICA (α).....	21
1.4. EQUAÇÃO DIFERENCIAL QUE DESCREVE A TRANSFERENCIAL DE CALOR	22
1.4.1. CONDIÇÕES DE CONTORNO DA EQUAÇÃO DO CALOR	25
1.5. TENSÕES	28
1.5.1. EQUAÇÕES CONSTITUTIVAS.....	29
1.5.2. EFEITOS DA TEMPERATURA EM SÓLIDOS E DEFORMAÇÕES INICIAIS.....	30
1.6. MÉTODOS NUMÉRICOS.....	31
1.7. MÉTODO DOS ELEMENTOS FINITOS	31
1.7.1. PROCEDIMENTO PADRÃO NA UTILIZAÇÃO DO MEF	33
1.8. EQUAÇÃO TRANSFERÊNCIA DE CALOR EM UM ELEMENTO FINITO	35
1.8.1. CONDIÇÕES DE CONTORNO	37
1.9. ESTADO PLANO DE TENSÃO E DEFORMAÇÃO.....	38
1.9.1. ESTADO PLANO DE TENSÃO	38
1.9.2. ESTADO PLANO DE DEFORMAÇÃO.....	40
1.10. ELEMENTO FINITO TRIANGULAR DE TRÊS NÓS.....	42
1.10.1. PROBLEMA TÉRMICO (REGIME PERMANENTE).....	45

1.10.2.	PROBLEMA TÉRMO-MECÂNICO.....	46
1.10.3.	ACOPLAMENTO DO PROBLEMA TÉRMO-MECÂNICO	48
2.	METODOLOGIA.....	49
3.	ALGORITMO EM FORTRAN.....	51
3.1.	MATRIZ DE RIGIDEZ DO ELEMENTO (TÉRMICO).....	52
3.2.	MATRIZ GLOBAL DE RIGIDEZ (TÉRMICO)	53
3.3.	CONDIÇÕES DE CONTORNO	54
3.4.	RESULTADO – TEMPERATURAS NODAIS	55
3.5.	ACOPLAMENTO TÉRMO-MECÂNICO	55
3.6.	MATRIZ DE RIGIDEZ DO ELEMENTO (MECÂNICA)	56
3.7.	MATRIZ GLOBAL DE RIGIDEZ (MECÂNICO)	57
3.8.	CONDIÇÕES DE CONTORNO	58
3.9.	RESULTADO – DESLOCAMENTOS NODAIS	59
3.10.	PÓS-PROCESSAMENTO	59
4.	ACOPLAMENTO TÉRMO-MECÂNICO UTILIZANDO ANSYS®.....	63
5.	VALIDAÇÃO DO ALGORITMO.....	64
6.	ESTUDO DE CASO - CENTRAL HIDRELÉTRICA DE ITAIPU (CHI).....	67
6.1.	DADOS GEOMÉTRICOS	71
6.1.1.	DADOS INSTRUMENTAIS.....	72
6.2.	MODELO MATEMÁTICO	75
6.3.	CONDIÇÕES DE CONTORNO	76
6.4.	MALHA	77
7.	RESULTADOS	79
8.	CONCLUSÕES	81
	REFERÊNCIAS BIBLIOGRÁFICAS.....	82
	APÊNDICE I – ARQUIVO .TXT DE ENTRADA DE DADOS	85
	APÊNDICE II – ALGORITMO COMPLETO	87
	APÊNDICE III – COMANDOS DE AJUSTES DE PLOTAGEM (GNU PLOT)	104
	ANEXO I – SUBROTINAS FORTRAN (INVERTER MATRIZES)	106

LISTA DE FIGURAS

Figura 1. Tipos de barragem de concreto: (a) barragem em arco; (b) barragem de terra e/ou enrocamento; (c) barragem de contrafortes; (d) barragem de gravidade	10
Figura 2. Vista lateral, frontal e superior de uma barragem em contraforte	11
Figura 3. Transferência de calor unidimensional por condução.....	16
Figura 4. Evaporação de água	17
Figura 5. Fissuras devido restrições ao movimento	19
Figura 6. Domínio bidimensional	22
Figura 7. Elemento diferencial.....	22
Figura 8. Divisão do contorno do domínio Ω	26
Figura 9. Equilíbrio do fluxo de calor no contorno	26
Figura 10. Notação das componentes de tensões em um bloco infinitesimal	28
Figura 11. Discretização do domínio	31
Figura 12. Exemplos de elementos finitos.....	32
Figura 13. Sólido adequado para um modelo no estado plano de tensões	38
Figura 14. Sólido adequado para um modelo no estado plano de deformação.	40
Figura 15. Elemento Triangular de Três Nós.....	42
Figura 16. Metodologia adotada.....	49
Figura 17. Diagrama Esquemático Geral da resolução do problema.....	51
Figura 18. Diagrama do algoritmo	52
Figura 19. Ilustração do pós-processamento	59
Figura 20. Criação de arquivos de texto de saída para plotar resultados	61
Figura 21. Comandos de ajustes no GNUPLOT.....	62
Figura 22. Plotagem de resultados utilizando o programa <i>GNUPLOT</i>	62
Figura 23. Procedimento para acoplamento de análises no Workbench (Ansys®).....	63
Figura 24. Campo de temperatura como condição de inicial da análise estática.....	63
Figura 25. Placa de alumínio e aço submetida à variação de temperatura e malha simplificada.....	64
Figura 26. Condições de restrição ao movimento	64
Figura 27. Comparação dos resultados de deslocamentos nodais do problema de validação	65
Figura 28. Placa engastada submetida a diferentes temperaturas nas bordas	65
Figura 29. Campo de Temperatura (Algoritmo).....	66
Figura 30. Campo de Temperatura (Ansys®).....	66
Figura 31. Campo de Deslocamento Vertical. (Algoritmo).....	66
Figura 32. Campo de Deslocamento Vertical. (Ansys®).....	66

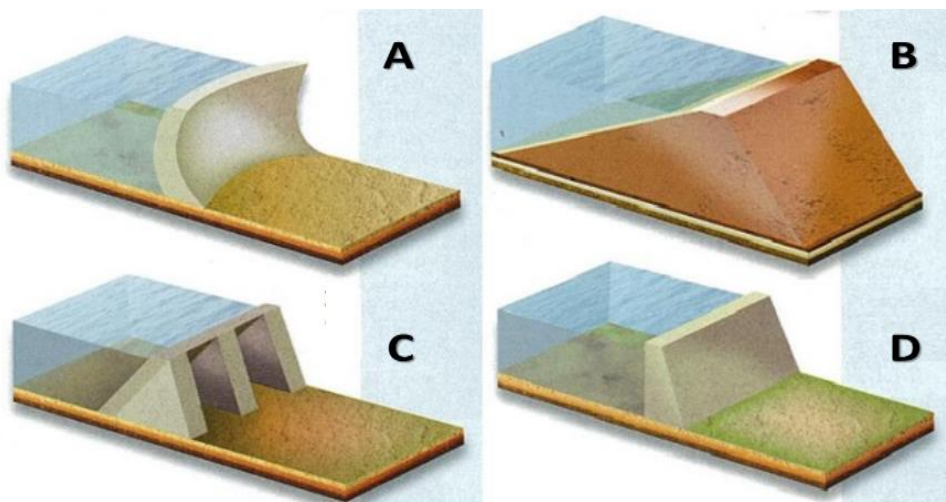
Figura 33. Campo de Deslocamento Horizontal. (Algoritmo).....	66
Figura 34. Campo de Deslocamento Horizontal. (Ansys®)	66
Figura 35. Projeto arquitetônico geral da CHI	67
Figura 36. Trecho E da CHI	68
Figura 37. Ilustração do funcionamento do pêndulo	69
Figura 38. Bloco E-6 da CHI	69
Figura 39. Variação da temperatura interna do bloco de contraforte E-6.....	70
Figura 40. Variação da Temperatura ambiente e deslocamento da crista da barragem	70
Figura 41. Vista Lateral do Bloco E-6.....	71
Figura 42. Localização dos termômetros do bloco E6.	72
Figura 43. Localização dos termômetros dos Bloco D-54 e do D-57	73
Figura 44. Variação do nível do reservatório e deslocamento da crista da barragem.	73
Figura 45. Variação do nível do reservatório e deslocamento da crista da barragem.	74
Figura 46. Geometria Bidimensional do bloco E6.....	75
Figura 47. Modelo matemático com condições de contorno pontuais	76
Figura 48. Ilustração do elemento retangular de quatro nós.....	77
Figura 49. Malha retangular uniforme de quatro nós	78
Figura 50. Malha triangular regular uniforme de três nós	78
Figura 51. Campo de Temperatura (verão) do algoritmo.....	79
Figura 52. Campo de Temperatura (verão) do programa ANSYS®.....	79
Figura 53. Campo de deslocamento na horizontal (verão) do algoritmo.....	79
Figura 54. Campo de deslocamento na horizontal (verão) do programa ANSYS®.....	79
Figura 55. Campo de Temperatura do algoritmo.....	80
Figura 56. Campo de Temperatura do programa ANSYS®.....	80
Figura 57. Campo de deslocamento na horizontal do algoritmo.....	80
Figura 58. Campo de deslocamento na horizontal do programa ANSYS®.....	80

INTRODUÇÃO

Barragem pode ser definida como um elemento estrutural construído transversalmente à direção de escoamento do curso de água de um rio, com o objetivo de elevar o nível de água a montante. As finalidades pela qual se constrói uma barragem são diversas, podendo ser: abastecimento de água, controle de inundações, irrigação, aproveitamento hidrelétrico, entre outros (SA, 2003).

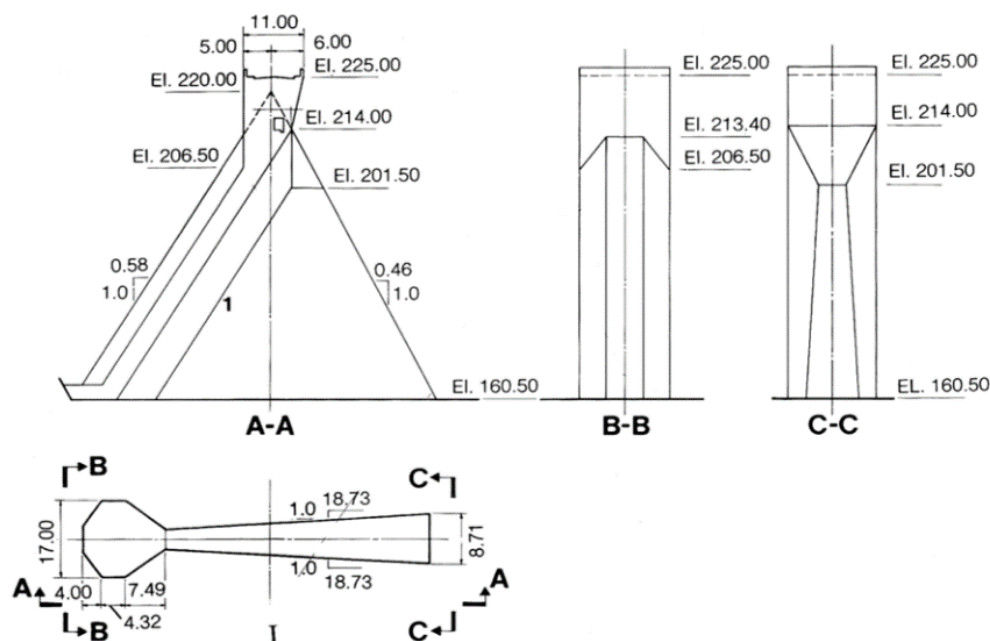
Em relação ao material, as barragens podem ser classificadas como barragem de terra/enrocamento e de concreto. Este último pode ser de gravidade, de gravidade aliviada, de arco ou contrafortes. A **Figura 1** ilustra os principais tipos de barragens de concreto.

Figura 1. Tipos de barragem de concreto: (a) barragem em arco; (b) barragem de terra e/ou enrocamento; (c) barragem de contrafortes; (d) barragem de gravidade



Fonte: Adaptado HICKMANN (2016).

Segundo JESUS (2011), em relação à geometria, as barragens de contrafortes apresentam uma laje de concreto com declive para montante, que é responsável pela impermeabilização e pela transmissão das cargas aos contrafortes. A estabilidade à ação da massa de água sobre o paramento de montante é assegurada através da mobilização do peso próprio. A utilização de contraforte, como elemento transversal à seção, contribui para a redução do volume de concreto, dissipação do calor de hidratação e para a diminuição das subpressões na base. A sua utilização é recomendável em vales largos, com baixa declividade, e com fundação de alta resistência para suportar as sobrecargas na fundação devido a reduzida área de contato. Na **Figura 2** é apresentado um perfil tipo de barragem de contraforte da Central Hidrelétrica de ITAIPU (CHI).

Figura 2. Vista lateral, frontal e superior de uma barragem em contraforte

Fonte: Adaptado ITAIPU (2017).

Em relação aos esforços, os principais carregamentos presentes em uma barragem são: o peso próprio, a pressão hidrostática, a subpressão e os esforços internos devido a variação térmica. Este último pode afetar o desempenho da barragem, diminuindo a resistência e conseqüentemente a durabilidade da estrutura.

Durante a fase de construção da barragem, a maior parte da variação térmica interna da estrutura ocorre devido reações químicas do concreto. Posteriormente, essa fonte de calor pode ser desprezada e a variação de temperatura passa a ter um comportamento essencialmente sazonal, onde a amplitude térmica ocorre devido fluxo de calor entre a superfície da barragem e o ambiente.

O Método dos Elementos Finitos (MEF) associado com a alta capacidade de processamento computacional se tornou uma ferramenta extraordinária para profissionais que atuam na área de simulação e análise de problemas de engenharia. Porém, além utilizar programas, é importante que o profissional entenda matemática e computacionalmente, o que está sendo processado por trás das interfaces dos programas. Deste modo, desenvolveu-se um código computacional em FORTRAN, baseado no Método dos Elementos Finitos capaz de simular o comportamento mecânico estrutural de um meio bidimensional submetido à transferência de calor por condução.

OBJETIVOS

OBJETIVO GERAL

Simular o comportamento térmico-estrutural de um bloco de contraforte da CHI, utilizando o Método dos Elementos Finitos.

OBJETIVOS ESPECÍFICOS

Para atingir o objetivo geral desta pesquisa, será necessário cumprir os seguintes objetivos específicos:

- a) Estudar a influência térmica nas estruturas de concreto massa;
- b) Determinar as propriedades térmicas do concreto da Barragem de ITAIPU;
- c) Realizar o levantamento de dados de projeto da barragem CHI e desenvolver a geometria do Bloco E-6 do Trecho E;
- d) Estabelecer as condições de contorno e calibragem do modelo matemático por meio de dados instrumentais;
- e) Desenvolver um código computacional com base no método dos elementos finitos capaz de simular o comportamento térmico de um meio bidimensional submetido a transferência de calor por condução
- f) Desenvolver um modelo matemático para simular o comportamento térmico estrutural da barragem de contrafortes da CHI.

1. REVISÃO BIBLIOGRÁFICA

De acordo com ZHU BUFANG (2004) as estruturas de concreto massa, que empregam grandes volumes de concreto durante a construção, são extremamente importantes, especialmente nas construções hidráulicas e hidroelétricas. Como as dimensões dessas estruturas são de grande porte, normalmente, é economicamente inviável a utilização de reforço, o que implica que todas as tensões devem ser suportadas unicamente pelo concreto.

É comum em barragens problemas provenientes da variação de temperatura, como fissuras por exemplo. Esta pode causar importante impacto no estado de tensão da estrutura. Por isso, um estudo aprofundado do comportamento térmico estrutural de concreto massa é muito importante considerando não somente a fase inicial de construção, mas também, toda a vida útil da estrutura (ZHU BUFANG, 2004).

TATIN et al (2015) desenvolveram um modelo físico-estatístico para melhorar a avaliação de deslocamentos de barragens, devido ao campo de temperatura. O modelo considerou, não somente a temperatura do ar, mas também a da água para estimar a temperatura a montante da estrutura. Essas considerações permitiram melhorar os valores médios do gradiente da temperatura dentro da estrutura, as quais são estimadas pela resolução analítica de um problema bidimensional de transferência de calor, por condução, com base na temperatura medida do ar e da água. Concluiu-se que o modelo heurístico de elementos finitos reduz a dispersão residual induzida pela temperatura da água. Mesmo utilizando valores médios de temperatura para alimentar o modelo, a melhoria é significativa tanto para a calibração como para a previsão.

JAAFAR et al (2007) realizam a análise térmica das barragens de concreto compactados a rolo (RCC) na concepção do desempenho durante a construção. Desenvolveu-se um algoritmo baseado em elementos finitos para a determinação de temperaturas dentro do corpo da barragem. Este foi aplicado em um problema real para se determinar as respostas térmica da estrutura em função do cronograma da concretagem da barragem. Concluiu-se que a mudança de horário e posicionamento de compactação do concreto pode otimizar a localização das zonas de temperatura máxima.

HICKMANN (2016) desenvolveu um modelo numérico utilizando elementos finitos para calibrar e validar parâmetros térmicos de um bloco de contraforte da Barragem da Usina Hidrelétrica de Itaipu sujeito à variação térmica sazonal. A modelagem térmica em barragem de contrafortes é realizada através de Equações Diferenciais Fracionárias. No trabalho verificou-se a importância de se explorar métodos

numéricos para a resolução de problemas de engenharia cujo domínio de estudo é irregular.

KAVAMURA (2005) realizou uma análise termo-estrutural bidimensional da Usina Hidrelétrica de Salto de Caxias utilizando o programa de simulação computacional ANSYS através do método dos elementos finitos. Foram avaliados os efeitos de carregamentos obtidos da análise térmica e do peso próprio da estrutura. Os resultados foram comparados com outros trabalhos realizados na mesma Usina.

MIRZABOZORG et al (2014) desenvolveu um modelo numérico utilizando o método dos elementos finitos da barragem em arco Karum. O modelo foi calibrado através de dados instrumentais. As propriedades térmicas do concreto foram obtidas por uma análise térmica transiente e os resultados foram comparados com os termômetros da barragem. A análise térmica considerou a temperatura do ar a jusante e a da água, a montante e a radiação solar. Para calibrar o modelo completo, considerou-se o campo de distribuição de temperatura, o peso próprio da estrutura, a pressão hidrostática e subpressão. O modelo permite utilizar dados atualizados da barragem para avaliações de segurança.

A autora COELHO (2016) analisou a construção de estruturas de concreto massa, abrangendo as metodologias aplicadas para a redução do gradiente térmico via MEF, como a construção em camadas. Antes de se utilizar o programa escolhido, ANSYS, foram realizadas validações com o método das diferenças finitas (MDF) e soluções analíticas para garantir o correto domínio da ferramenta. Após obter as curvas de temperatura transiente, examinou-se as tensões em alguns pontos da estrutura analiticamente, com auxílio do MATLAB. Posteriormente, aplicou-se curvas de fluência do concreto. Todos os métodos se mostraram eficientes no cálculo de tensões.

LÉGER et al (2009) estudaram os deslocamentos termomecânicos utilizando o pêndulo em barragens de gravidade através de um modelo que compara uma barragem simplificada com elemento de viga e um modelo de deslocamento estático. Realizou-se comparações entre as análises de transferência de calor de seções típicas de barragens de gravidade e análises de elementos finitos bidimensional.

KHAN (2001) realizou uma análise termomecânica de uma barragem CCR (Concreto compactado a rolo) utilizando o programa ANSYS®, na fase de construção como estudo de caso. O autor afirma que a construção pelo método rampado, que está sendo difundido, precisa ser melhorado. Os resultados da análise em termos de temperaturas ou tensões foram combinados com os resultados da análise estrutural, para avaliar o comportamento de corpo da barragem com todas as cargas.

1.1. TRANSFERÊNCIA DE CALOR

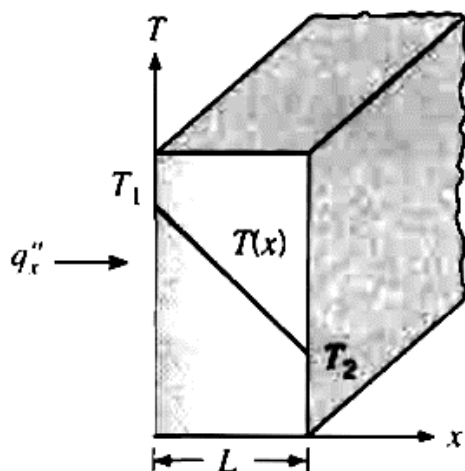
A dedução apresentada a seguir é baseada em INCROPERA et al (1998).

O calor é definido como a forma de energia que se transfere de um sistema para outro devido a diferença de temperatura entre eles. A taxa de transferência é a quantidade de calor transferida por unidade de tempo. Sempre que existir uma diferença de temperatura, haverá um ou mais processos de transferência de calor. Quando existe um gradiente de temperatura estacionário, ou seja, sem variações no tempo, em um meio sólido ou fluído, considera-se a condução para se referir à transferência de calor que ocorre entre as superfícies. Outro modo de transferência de calor é a radiação, onde as superfícies de temperatura finita emitem energia sob a forma de ondas eletromagnéticas, que pode ocorrer também, na ausência de um meio intermediário entre duas superfícies com diferentes temperaturas.

1.1.1. CONDUÇÃO

A concepção de condução está relacionada com a atividade molecular. A condução deve ser vista como a transferência de energia de uma partícula com mais energia para outra com menos energia devido a interação entre elas. Altas temperaturas são associadas com altas energias moleculares, e quando uma molécula adjacente entra em contato, uma transferência de energia, da molécula com maior energia para a de menor energia ocorre. Na presença de um gradiente de temperatura, a transferência de energia através da condução deve ocorrer na direção decrescente de temperatura. É possível quantificar o processo de transferência de calor. Essas equações são utilizadas para quantificar a quantidade de energia que é transferida por unidade de tempo. Para a condução de calor, a equação da taxa é conhecida como a Lei de Fourier.

Considerando uma parede plana cuja transferência de calor ocorre em apenas uma direção, conforme a taxa pode ser escrita como é apresentado na Eq. (1).

Figura 3. Transferência de calor unidimensional por condução

Fonte: INCROPERA (1998)

$$q_x'' = -k_x \frac{dT}{dx} \quad (1)$$

O fluxo de calor q_x'' (w/m^2) é a taxa de transferência na direção x por unidade de área perpendicular à direção de transferência, e é proporcional ao gradiente de temperatura (dT/dx). O parâmetro k_x representa a propriedade de transferência e é característico do material da parede. O sinal negativo indica que a transferência ocorre na direção contrária à maior temperatura. Em condições de estado estacionário, onde a distribuição de temperatura é linear o gradiente de temperatura pode ser calculado através da Eq.(2).

$$\frac{dT}{dx} = \frac{T_2 - T_1}{L} \quad (2)$$

Sendo T_1 é a maior temperatura, T_2 é a menor temperatura e L é a distância entre as superfícies.

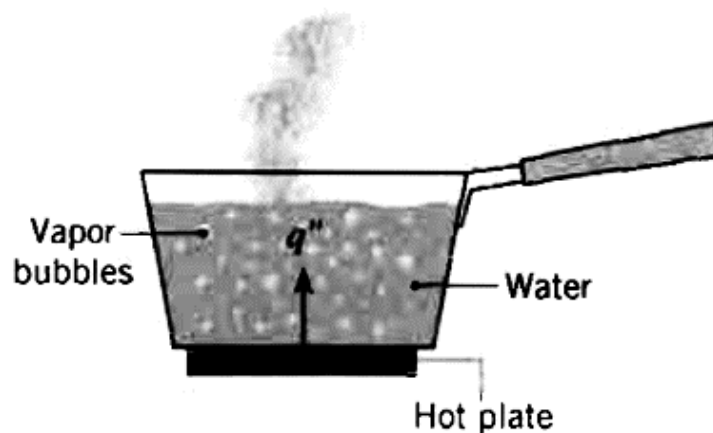
De acordo com a Eq.(2), o fluxo pode ser reescrito conforme a Eq.(3).

$$q_x'' = -k_x \frac{dT}{dx} = -k \frac{\Delta T}{L} \quad (3)$$

1.1.2. CONVECÇÃO

A transferência de calor por convecção é composta por dois mecanismos. Juntamente com a transferência de energia devido a movimentação molecular (difusão), ocorre também a transferência de massa, ou movimentação macroscópica do fluido. Essa movimentação do fluido está associada à movimentação coletiva das moléculas. A transferência total é então uma superposição de transporte de energia devido ao movimento randômico das moléculas e o movimento de massa do fluido. No caso de um sistema formado por um material sólido qualquer cuja temperatura esteja acima da temperatura do ar ao seu redor, a transferência de calor por convecção que ocorre no sentido vertical. O ar que está em contato com o material sólido sofre um aumento de temperatura devido a condução de calor, e conseqüentemente reduz a densidade. Essas moléculas se tornam mais leves que o ar ao redor e sofrem empuxo, uma força vertical para cima. Assim, o ar quente sobe e o espaço é preenchido imediatamente pelo ar frio ao redor. A **Figura 4** ilustra um exemplo de convecção onde o vapor da água fervendo sobe devido à baixa densidade em relação ao ar ao redor.

Figura 4. Evaporação de água



Fonte: INCROPERA (1998)

O processo da taxa de transferência de calor por convecção natural pode ser escrito com a Eq.(5).

$$q_x'' = h(T_s - T_\infty) \quad (4)$$

Onde q'' é a taxa de convecção de calor, e é proporcional a diferença entre a temperatura da superfície do sólido e do ar ao redor, e h é o coeficiente de transferência de calor por convecção.

1.1.3. RADIAÇÃO

A radiação térmica é a energia emitida pelo material cuja temperatura é diferente de zero. A energia do campo de radiação é transportada por ondas eletromagnéticas. Diferente da transferência de energia por condução ou convecção, a radiação não requer a presença de um material como intermédio, sendo mais eficiente no vácuo.

A radiação emitida de uma superfície originada da energia térmica da fronteira pela superfície do material e a taxa em que a energia é liberada por unidade de área é chamada de poder de emissão de superfície. Há um limite superior de capacidade de emissão que é descrito pela Lei de Stefan-Boltzmann, conforme a Eq.(5).

$$E_b = \sigma T_s^4 \quad (5)$$

Onde T_s é a temperatura absoluta e σ é a constante de Stefan-Boltzmann igual a $\sigma = 5.67 * 10^{-8} \left(\frac{W}{m^2} K^4 \right)$, conhecida como radiador ideal ou material negro.

O fluxo emitido por uma superfície real é menor que a de um *blackbody* e a mesma temperatura é dada pela Eq.(6).

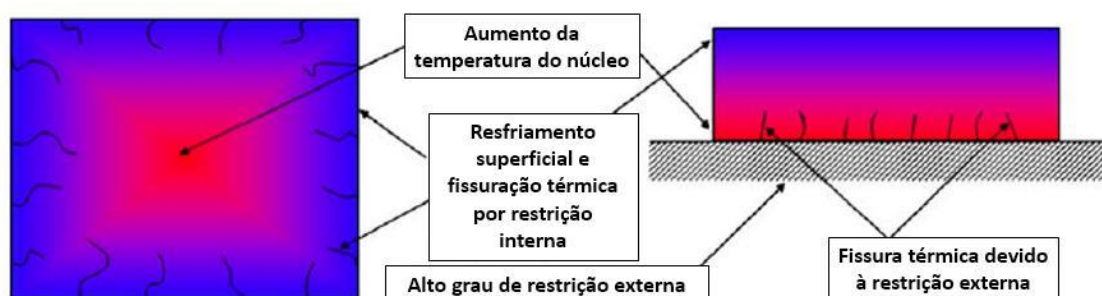
$$E_b = \varepsilon \sigma T_s^4 \quad (6)$$

Onde ε é a propriedade radioativa da superfície chamada emissividade, cujos valor varia de 0 a 1. Essa propriedade indica o quão eficiente a superfície emite a energia em relação ao corpo negro.

1.2. TENSÕES TÉRMICAS

Segundo ARAÚJO (2013), a reação de hidratação do cimento é exotérmica, ou seja, libera calor no meio. Devido à baixa capacidade de condutividade térmica do concreto, o calor gerado na reação química entre o cimento e a água, eleva a temperatura interna da massa. No caso de estruturas de concreto massa, esse efeito é ainda maior, uma vez que o efeito é proporcional a quantidade de concreto. O calor, transferido por condução, é dissipado no meio ambiente ao alcançar as superfícies da peça. Devido à transferência de calor, surgem gradientes de temperatura que ocasionam tensões de tração e de compressão no concreto, conforme a **Figura 5**.

Figura 5. Fissuras devido restrições ao movimento



Fonte: Araújo (2013)

Por estar em contato com o meio externo, a superfície da peça resfria mais rápido e sofre contração térmica. Por outro lado, devido ao confinamento, o núcleo conserva altas temperaturas, introduzindo à peça, esforços de tração. Esse fenômeno pode causar fissuras no concreto e comprometer sua durabilidade. Esse é um problema típico de deformações impostas internamente e é independente de carregamentos externos. A inércia térmica do concreto, consequência da baixa condutividade, impede a rápida transferência de calor, ocasionando diferentes temperaturas em vários pontos do elemento estrutural, causando tensões de compressão no interior e tensões de tração nas superfícies.

No caso de estruturas de grande volume, como é o caso das barragens, há vários estudos sobre transferência de calor e técnicas construtivas como a concretagem em camadas de pequena altura, a pré-refrigeração e a pós-refrigeração do concreto.

1.3. PROPRIEDADES DO CONCRETO

A caracterização das propriedades térmicas dos materiais é uma das fases mais importantes devido a influência que esta tem em relação ao comportamento da estrutura.

No caso de estruturas de concreto massa, a hidratação do cimento e a baixa dissipação de calor, resultam no aumento da temperatura do concreto na fase inicial. No caso das barragens, o controle de temperatura do concreto através da seleção do cimento e o uso gelo durante a concretagem, garantem condições adequadas de estabilidade da estrutura, evitando patologias de origem térmica (CASTILHO, 2013).

Normalmente, as propriedades do concreto como a densidade, módulo de elasticidade e a resistência à compressão, são estimados através de estudos laboratoriais. No entanto, não é comum a realização de ensaios para obtenção das propriedades térmicas do concreto, como condutividade térmica e calor específico, sendo necessário recorrer a valores indicados na literatura para a realização de análises térmicas (BRAGA, 2004). As principais propriedades que são utilizadas para a análise termo-mecânica neste trabalho são: Módulo de elasticidade (E), coeficiente de Poisson (ν), condutividade térmica (k), e o coeficiente de dilatação térmica (α).

1.3.1. MÓDULO DE ELASTICIDADE (E)

O Módulo de Elasticidade ou Módulo de Young é uma grandeza proporcional à rigidez de um material quando este é submetido a uma tensão externa de tração ou compressão. Basicamente, é a razão entre a tensão aplicada e a deformação sofrida pelo corpo quando o comportamento é linear, conforme a Eq.(7).

$$E = \frac{\sigma}{\varepsilon} \quad (7)$$

Sendo E o Módulo de elasticidade, σ a tensão e ε a deformação.

1.3.2. COEFICIENTE DE POISSON (ν)

O coeficiente de Poisson ou Razão de Poisson, mede a deformação transversal, em relação à direção longitudinal de aplicação da carga, de um material homogêneo e isotrópico. A razão é estabelecida entre as deformações ortogonais, conforme a Eq.(8)

$$\nu = - \frac{\varepsilon}{\varepsilon_{\text{perpend}}} \quad (8)$$

Sendo ν o coeficiente de Poisson, ε a deformação na direção do carregamento e $\varepsilon_{\text{perpend}}$ a deformação na direção perpendicular ao carregamento.

1.3.3. CONDUTIVIDADE TÉRMICA (K)

Pode ser definida como a taxa de transferência de calor por meio de uma unidade de comprimento de um material por unidade de área, considerando diferentes temperatura. Esta propriedade quantifica a tendência de o material conduzir calor (ÇENGEL, 2003).

1.3.4. COEFICIENTE DE DILATAÇÃO TÉRMICA (α)

O coeficiente de dilatação térmica define a deformação correspondente à variação de 1°C de um material. No caso do concreto armado, para variações normais de temperatura, o valor recomendado é de 10-5/°C (NBR 6118, item 8.2.3).

Em 1976, as empresas PROMON e ALTO PARANÁ, através de consultoria prestada à CHI durante a fase de construção, adotaram as propriedades térmicas do concreto apresentadas na **Tabela 1**, para a realização de análises de tensões de origem térmica utilizando MEF. Esses mesmos parâmetros serão utilizados neste trabalho.

Tabela 1. Propriedades térmicas adotados segundos (PROMON e ALTO PARANA, 1976)

Propriedade	Concreto		Fonte
Módulo de Elasticidade (E)	35.000	MPa	PROMON
Coeficiente de Poisson (ν)	0,16	-	NBR6118
Condutividade Térmica (k)	1,8492	W.m ⁻¹ .K ⁻¹	PROMON
Coeficiente de Dilatação Térmica (α)	0,0001	/°C	NBR6118

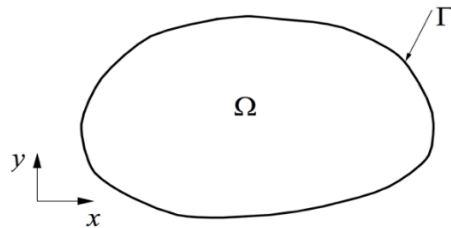
Fonte: ITAIPU (2017).

1.4. EQUAÇÃO DIFERENCIAL QUE DESCREVE A TRANSFERENCIAL DE CALOR

A dedução apresentada a seguir é baseada em SOUZA (2003).

Considerando um domínio bidimensional com domínio Ω e contorno Γ , referenciado em um sistema de coordenadas cartesianas (x, y) , conforme é apresentado na **Figura 6**.

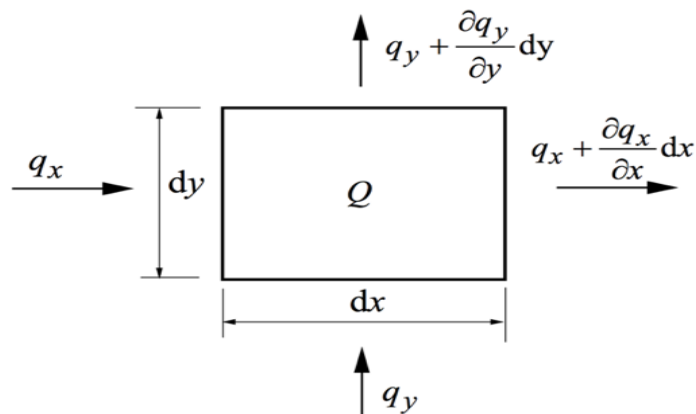
Figura 6. Domínio bidimensional



Fonte: Souza (2003)

Seja $Q(x, y)$ a fonte da taxa de geração de calor interna (calor por unidade de volume e tempo), e $q_x(x, y)$ e $q_y(x, y)$, as componentes do fluxo de calor (por unidade de área e tempo) em um ponto qualquer dentro do domínio. A equação que governa o fenômeno de condução de calor em um meio bidimensional em equilíbrio, ou seja, regime estacionário, sem variação no tempo, é chamada de Lei de Fourier e pode ser deduzida considerando um elemento diferencial de lados dx e dy , com o fluxo de calor atravessando o contorno do domínio, conforme mostra a **Figura 7**.

Figura 7. Elemento diferencial



Fonte: Souza (2003)

Considerando-se que não há perda de energia, a taxa de calor gerada no corpo é igual a $Qdxdy$ e a Eq.(9) deve ser satisfeita.

$$Qdxdy + q_x dy + q_y dx = \left(q_x + \frac{\partial q_x}{\partial x} \right) dy + \left(q_y + \frac{\partial q_y}{\partial y} \right) dx \quad (9)$$

Simplificando a Eq.(9) e dividindo-a por $dxdy$, chega-se na equação que governa o problema estacionário de condução de calor, conforme mostra a Eq.(10).

$$-\frac{\partial q_x}{\partial x} - \frac{\partial q_y}{\partial y} + Q = 0, \text{ no domínio } \Omega \quad (10)$$

Existem várias formas matemáticas de simplificar essa expressão, e algumas delas são apresentadas na Eq. (11).

$$- \operatorname{div} q + Q = 0 \text{ ou } - \nabla q + Q = 0 \quad (11)$$

Onde $\nabla = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix}$ denota o operador diferencial nabla de forma que:

$$\nabla q = \left\langle \frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right\rangle \begin{Bmatrix} q_x \\ q_y \end{Bmatrix} = \frac{\partial q_x}{\partial x} - \frac{\partial q_y}{\partial y} = \operatorname{div} q \quad (12)$$

Em relação ao fluxo de calor unidimensional, este é proporcional à taxa de variação de temperatura naquela direção. Esse comportamento físico é chamado de Lei de Fourier, conforme mostra a Eq.(13).

$$q_x = k_x \frac{\partial T}{\partial x} \quad (13)$$

sendo k_x é o coeficiente de condutividade térmica na direção x .

No caso bidimensional, o vetor de fluxo calor é função do gradiente de temperatura T .

$$q = -k \nabla T \quad (14)$$

Onde k é a matriz de condutividade térmica, mostrada na Eq.(15).

$$k(x, y) = \begin{pmatrix} k_{xx}(x, y) & k_{xy}(x, y) \\ k_{yx}(x, y) & k_{yy}(x, y) \end{pmatrix} \quad (15)$$

E ∇T é gradiente de temperatura, ilustrado na Eq.(16).

$$\nabla T = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} T = \begin{pmatrix} \frac{\partial T}{\partial x} \\ \frac{\partial T}{\partial y} \end{pmatrix} = \text{grad}T \quad (16)$$

Dessa forma, a Eq.(14) pode ser reescrita na forma que é mostrada na Eq.(17).

$$\begin{pmatrix} q_x \\ q_y \end{pmatrix} = \begin{pmatrix} k_{xx} & k_{xy} \\ k_{yx} & k_{yy} \end{pmatrix} \begin{pmatrix} \frac{\partial T}{\partial x} \\ \frac{\partial T}{\partial y} \end{pmatrix} \quad (17)$$

Sendo:

$$q_x = \left(k_{xx} \frac{\partial T}{\partial x} + k_{xy} \frac{\partial T}{\partial y} \right) \text{ e } q_y = - \left(k_{xy} \frac{\partial T}{\partial x} + k_{yy} \frac{\partial T}{\partial y} \right) \quad (18)$$

Substituindo a Eq.(18) na Eq.(10), obtém-se a seguinte expressão:

$$\frac{\partial}{\partial x} \left(k_{xx} \frac{\partial T}{\partial x} + k_{xy} \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial y} \left(k_{xy} \frac{\partial T}{\partial x} + k_{yy} \frac{\partial T}{\partial y} \right) + Q = 0 \quad (19)$$

Para fins de simplificação dos cálculos, pode-se adotar algumas hipóteses: Coincidência entre as direções cartesianas x e y e as direções principais do material. Isso implica que k_{xy} seja igual a zero ($k_{xy} = 0$); Material com propriedades isotrópicas. Resulta que k_{xx} seja igual a k_{yy} ($k_{xx} = k_{yy} = k$); Material homogêneo, ou seja, a condutividade térmica não depende da direção (x, y).

Dessa forma, a matriz de condutividade fica como é mostrado na Eq.(20).

$$k = k(x, y) = \begin{bmatrix} k(x, y) & 0 \\ 0 & k(x, y) \end{bmatrix} = k(x, y) \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = k(x, y)I \quad (20)$$

Onde I é a matriz de identidade de ordem 2.

Aplicando as hipóteses descritas acima na Eq.(19), obtém-se a Eq.(21), conhecida como a Equação de Poisson que governa vários problemas de campo na engenharia.

$$k \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) + Q = 0 \quad (21)$$

Reescrevendo a Eq.(21) de forma compacta, tem-se:

$$k \nabla^2 T + Q = 0 \quad (22)$$

sendo $\nabla^2 T$ é igual a

$$\nabla^2 T = \left\langle \frac{\partial}{\partial x} \quad \frac{\partial}{\partial y} \right\rangle \left\{ \begin{array}{c} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{array} \right\} T = \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \quad (23)$$

Para o caso particular onde $Q = 0$, ou seja, sem fonte calor, a Eq.(24) chamada de Equação de Laplace

$$k \nabla^2 = 0 \text{ ou } \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \quad (24)$$

1.4.1. CONDIÇÕES DE CONTORNO DA EQUAÇÃO DO CALOR

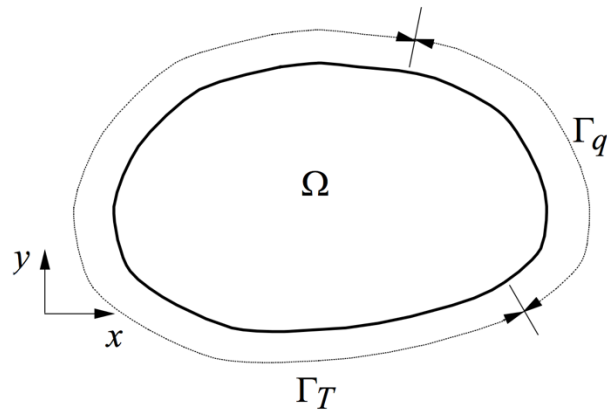
Em geral há três tipos de condição de contorno para o problema de condução de calor: imposição de temperatura; imposição de fluxo de calor; imposição de temperatura e fluxo de calor.

Geralmente, os dados instrumentais disponíveis para a modelagem e simulação de condução de calor em problemas reais são provenientes de termômetros. Por isso, neste trabalho abordou-se somente as imposições de temperatura em problemas térmicos.

Considerando o contorno Γ do domínio Ω ilustrado na **Figura 6**, de tal forma que onde há condição de fluxo de calor imposto, não há imposição de temperatura e vice-versa, conforme a Eq.(25).

$$\begin{aligned} \Gamma_T \cup \Gamma_q &= \Gamma \\ \Gamma_T \cap \Gamma_q &= \emptyset \end{aligned} \quad (25)$$

A condição descrita na Eq.(25) é ilustrada na Figura 8.

Figura 8. Divisão do contorno do domínio Ω 

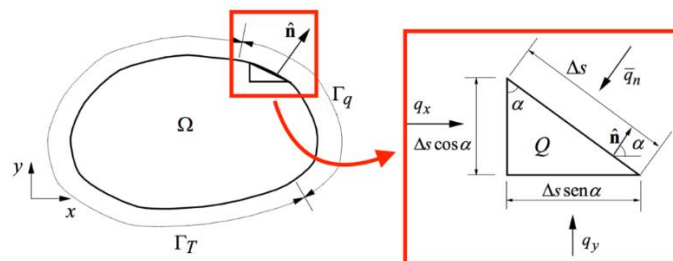
Fonte: Souza (2003)

A região de contorno cuja condição é dada por Γ_T corresponde ao tipo mais simples de condição de contorno. Consiste basicamente na especificação do valor da temperatura nessa região, conforme a Eq. (26).

$$T = T_{contorno}, em \Gamma_T \quad (26)$$

Onde $T_{contorno}$ é a temperatura conhecida no contorno.

No caso da imposição de fluxo de calor, considera-se o equilíbrio de fluxo em um elemento infinitesimal na região Γ_q , conforme a **Figura 9**.

Figura 9. Equilíbrio do fluxo de calor no contorno

Fonte: Adaptado Souza (2003)

Na **Figura 9** é mostrado o detalhe de um elemento diferencial com as componentes do fluxo de calor que atuam no contorno do elemento. O vetor \hat{n} é:

$$\hat{n} = \begin{Bmatrix} \hat{n}_x \\ \hat{n}_y \end{Bmatrix} = \begin{Bmatrix} \cos \alpha \\ \sin \alpha \end{Bmatrix} \quad (27)$$

É o vetor unitário da superfície do contorno, e α é o ângulo normal que esse vetor forma com o eixo das abscissas, Δ_s é o comprimento da face do elemento triangular referente ao contorno e \bar{q}_n é o valor conhecido do fluxo normal à superfície no contorno Γ_q .

O equilíbrio do fluxo de calor é estabelecido se a Eq.(28) for respeitada.

$$Q \frac{1}{2} \Delta_s^2 \cos \alpha \sin \alpha + q_x \Delta_s \cos \alpha + q_y \Delta_s \sin \alpha + \bar{q}_n \Delta_s = 0 \quad (28)$$

Dividindo a Eq.(28) por Δ_s , tem-se:

$$Q \frac{1}{2} \Delta_s \cos \alpha \sin \alpha + q_x \cos \alpha + q_y \sin \alpha + \bar{q}_n = 0 \quad (29)$$

Fazendo o limite da Eq.(29) quando Δ_s tende a zero, tem-se:

$$q_x \cos \alpha + q_y \sin \alpha + \bar{q}_n = q_x \hat{n}_x + q_y \hat{n}_y + \bar{q}_n = 0 \quad (30)$$

Reescrevendo de forma compacta:

$$-q^T \hat{n} = \bar{q}_n, \text{ em } \Gamma_q \quad (31)$$

1.5. TENSÕES

De acordo com ASGHAR (2005), as solicitações internas são produzidas em um corpo quando este é submetido a um carregamento externo. Essas forças internas são caracterizadas em termos de tensões. A tensão em um ponto P qualquer de um corpo é definido com base em um plano da seção transversal que passa pelo ponto P, considerando uma força por unidade de área em um elemento diferencial. Denotando a área desse elemento diferencial por ΔA e a força por ΔF , o vetor de tensões no ponto P para o plano definido cuja unidade normal ao plano é n , pode ser escrito conforme a Eq.(32).

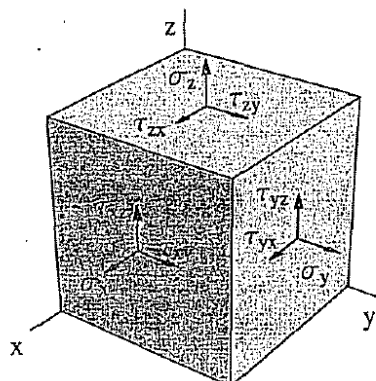
$$t_n(P) = \lim_{\Delta A \rightarrow 0} \frac{\Delta F}{\Delta A} \quad (32)$$

As tensões normais e cisalhantes são escritas considerando as forças nas direções normal (ΔF_n) e tangencial (ΔF_t) do plano, conforme a Eq.(33).

$$\begin{aligned} \sigma_n(P) &= \lim_{\Delta A \rightarrow 0} \frac{\Delta F_n}{\Delta A} \\ \tau_n(P) &= \lim_{\Delta A \rightarrow 0} \frac{\Delta F_t}{\Delta A} \end{aligned} \quad (33)$$

A definição de tensão está associada a um plano de seção transversal em termos da normal unitária n . A quantidade de seções transversais que passa por um ponto é infinita. Por isso, na prática, utiliza-se o sistema de coordenadas cartesianas para se definir os planos de seção transversal, que podem ser ilustrados um bloco infinitesimal, conforme a **Figura 10**.

Figura 10. Notação das componentes de tensões em um bloco infinitesimal



Fonte: Asghar (2005)

Considerando os vetores de tensões como uma matriz 3x3, obtém-se o tensor de tensões, conforme a Eq.(34).

$$S = \begin{pmatrix} \sigma_x & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_y & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_z \end{pmatrix} \quad (34)$$

1.5.1. EQUAÇÕES CONSTITUTIVAS

A tensão e a deformação podem ser relacionadas pela equação constitutiva. A forma mais simples dessa equação é a da elasticidade linear, que é a generalização da Lei de Hooke. Assumindo que a tensão e a deformação são proporcionais, a equação constitutiva assume a forma apresentada na Eq.(35).

$$\sigma = C \varepsilon \quad (35)$$

Se as propriedades do material são simétricas em qualquer plano, então, este recebe o nome de isotrópico e necessita apenas de duas constantes independentes para definir completamente as equações constitutivas, o módulo de elasticidade e o coeficiente de Poisson, conforme a Eq.(36).

$$\begin{pmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{pmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} \end{pmatrix} \begin{pmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{pmatrix} \quad (36)$$

A relação inversa é escrita conforme a Eq.(37).

$$D = \begin{pmatrix} \frac{1}{E} & -\frac{\nu}{E} & -\frac{\nu}{E} & 0 & 0 & 0 \\ -\frac{\nu}{E} & \frac{1}{E} & -\frac{\nu}{E} & 0 & 0 & 0 \\ -\frac{\nu}{E} & -\frac{\nu}{E} & \frac{1}{E} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G} \end{pmatrix} \quad (37)$$

Sendo $G = \frac{E}{2(1+\nu)}$.

1.5.2. EFEITOS DA TEMPERATURA EM SÓLIDOS E DEFORMAÇÕES INICIAIS

De acordo com ASGHAR (2005), assume-se inicialmente que não há carregamento e deformações no corpo. No entanto, há uma situação onde ocorre deformações que devem ser consideradas. A situação mais comum é causada pela deformação em um sólido devido a variação de temperatura.

O aumento da temperatura (ΔT) resulta em uma deformação que depende do coeficiente térmico de expansão (α) do material. A mudança de temperatura não causa tensões cisalhantes. Assim, o vetor de deformação inicial devido a variação de temperatura pode ser escrito conforme a Eq.(38).

$$\epsilon_0 = \begin{pmatrix} \alpha\Delta T \\ \alpha\Delta T \\ \alpha\Delta T \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (38)$$

A tensão causada pela diferença total de deformação e deformação inicial é apresentada na Eq.(39).

$$\sigma = C(\epsilon - \epsilon_0) \quad (39)$$

Sendo ϵ_0 o vetor de deformações iniciais. Invertendo a Eq. (39) obtém-se uma relação da deformação em função da tensão, conforme a Eq.(40).

$$\epsilon = D\sigma - \epsilon_0 \quad (40)$$

A Eq.(40) exhibe a tensão solicitante que um matéria apresenta devido a variação de temperatura do meio.

1.6. MÉTODOS NUMÉRICOS

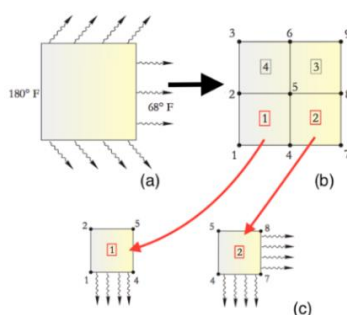
Equações diferenciais que representam um fenômeno físico, onde se impõem condições iniciais de contorno, precisam de uma solução para sua aplicação prática. O ideal é a solução exata, baseada em métodos algébricos diferenciáveis, porém, dependendo da complexidade do problema, torna-se impraticável. Por esse motivo, recorre-se a métodos numéricos, obtendo-se soluções aproximadas. Dentre os diversos métodos existem, o Método dos Elementos Finitos é um dos mais populares em análises de engenharia (HICKMANN, 2016).

1.7. MÉTODO DOS ELEMENTOS FINITOS

Segundo JAIME e MOLINA (2010), o MEF (Método dos Elementos Finitos) é um método de solução aproximada de equações diferenciais ordinárias ou parciais, muito utilizado na engenharia devido a sua generalidade e a facilidade de se aplicar em domínios geometricamente complicados. Esse método possibilita a simulação de situações reais em um domínio discreto, cujo limite infinitesimal tende ao contínuo.

O MEF consiste em dividir o problema real em partes menores de dimensões finitas, de forma que, o conjunto de todos os subdomínios seja igual ao original, conforme a **Figura 11**. Em cada subdomínio são utilizadas funções algébricas que calculam aproximações locais do fenômeno físico. Quanto maior for o número de subdivisões, mais próximo o resultado estará do valor exato. Por tornar o meio contínuo em várias partes discretas finitas, os subdomínios são chamados de elementos finitos. A **Figura 11 (c)** mostra exemplos de elementos finitos.

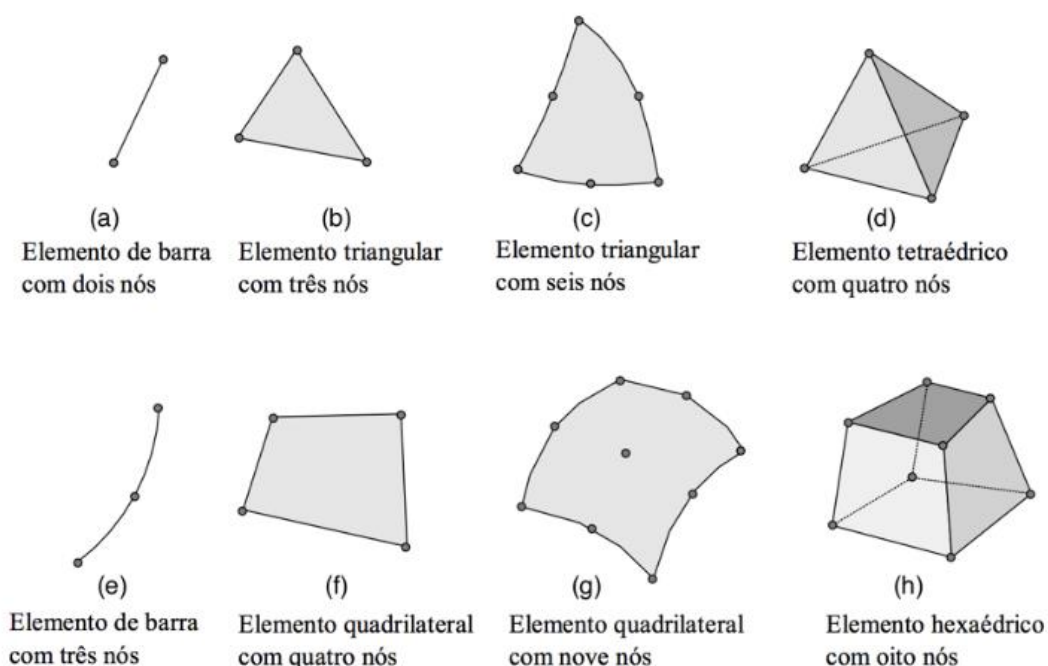
Figura 11. Discretização do domínio



Fonte: JAIME; MOLINA (2010).

Segundo LOGAN (2006), os elementos finitos são classificados de acordo com a sua forma geométrica, funções de aproximação e pelos tipos de problemas para os quais foram desenvolvidos, conforme ilustra a **Figura 12**. Cada elemento possui uma quantidade determinada de pontos nodais (nós), que são os responsáveis pela comunicação entre os elementos vizinhos. Os nós compartilhados por diferentes elementos adjacentes terão o mesmo valor independentemente dos elementos.

Figura 12. Exemplos de elementos finitos



Fonte: SOUZA (2003).

O conjunto de elementos finitos e pontos nodais que formam o corpo de um domínio discretizado recebe o nome de malha (SOUZA, 2003).

As informações de cada elemento são alocadas de forma matricial formando o sistema global de equações (matriz global), cuja solução fornece os valores das incógnitas nos pontos nodais, sendo as mesmas calculadas em função dos valores dos nós conhecidos (condição de contorno). No interior de cada elemento, bem como no limite do domínio, a varável de campo é descrita através de interpolação entre os nós (ASGHAR, 2005).

Segundo Jaime e Molina (2010), as características primárias de um elemento finito são inseridas na matriz de rigidez. No caso de um elemento estrutural, a matriz de rigidez contém informações da geometria e das propriedades do material, por exemplo,

a resistência e a deformação deste, quando submetido a um estado de carga. As deformações produzidas podem incluir efeitos como tração, compressão, flexão, torção e cisalhamento. No caso problemas não estruturais, como transmissão de calor, por exemplo, também se utiliza o termo de matriz de rigidez, uma vez que esta aloca os coeficientes que representam a resistência do elemento à mudança de estado, quando submetido a influência externa.

Há três métodos que podem ser utilizados para se obter as equações de elementos finitos de um problema: O método direto, o método variacional e o método dos resíduos ponderado. O método direto fornece uma ideia física muito clara do MEF e é excelente para se iniciar os estudos de elementos finitos. Porém, seu uso é limitado a problemas unidimensionais. O método variacional requer a existência de um funcional, cuja minimização resulta nas equações diferenciais que descrevem os fenômenos físicos. Dentre os métodos dos resíduos ponderados, o método de Galerkin é um dos mais populares e é o que mais se adequa a maioria dos problemas (KIM, N. H. SANKAR, 2011).

1.7.1. PROCEDIMENTO PADRÃO NA UTILIZAÇÃO DO MEF

Segundo Jaime e Molina (2010), as análises matemáticas para simular problemas de engenharia podem ser divididas em três etapas:

- Pré-processamento;
- Processamento;
- Pós-processamento.

1.7.2. PRÉ-PROCESSAMENTO

Fase onde se define o modelo. Nessa fase inclui-se:

- Definição do domínio geométrico do problema;
- Definição do tipo de elemento;
- Propriedades dos materiais dos elementos;
- Propriedades geométricas dos elementos (comprimento e área);
- Conectividades dos elementos (Malha);
- Restrições físicas (condições de contorno);
- Cargas externas.

O pré-processamento é a fase mais importante de uma análise física numérica, principalmente quando se utiliza softwares disponíveis no mercado. Definições de parâmetros equivocados, mesmo com a convergência numérica, resultará em valores errados, até mesmo sem significado físico.

1.7.3. PROCESSAMENTO

Durante a fase de solução, os softwares de elemento finito montam as equações algébricas governantes de forma matricial e calculam os valores desconhecidos das variáveis primárias do campo do problema. Os valores calculados são utilizados na substituição regressiva para avaliar as variáveis adicionais, derivadas secundárias, como forças de reação, tensões do elemento, fluxo de calor, velocidade e acelerações de movimento (JAIME E MOLINA, 2010).

É comum um modelo matemático de elementos finitos ser representado por milhares de equações em um problema. Nesses casos, são utilizadas técnicas especiais de solução para reduzir o armazenamento de dados e o tempo computacional. Em problemas lineares, por exemplo a estática, o algoritmo normalmente, é muito eficiente baseado em um procedimento de eliminação de Gauss (BATHE, 1977).

1.7.4. PÓS-PROCESSAMENTO

Na fase de pós-processamento, analisa-se os resultados da solução. Normalmente, os softwares associados a esse processo contêm rotinas sofisticadas utilizadas para organizar, imprimir e exibir os resultados selecionados da solução. Entre os resultados operacionais que são utilizados nas análises de resultados estão:

- Tensões internas;
- Equilíbrio estático;
- Deslocamentos ou deformações estruturais;
- Determinação de fatores de segurança;
- Respostas dinâmicas;
- Campos de temperatura.

O principal objetivo do pós-processamento é a apresentação de resultados legítimos, que após análise são utilizados para tomadas de decisão de engenharia.

1.8. EQUAÇÃO TRANSFERÊNCIA DE CALOR EM UM ELEMENTO FINITO

O desenvolvimento deste capítulo é baseado em JAIME e MOLINA (2010).

Considerando a Eq.(21) para desenvolver um elemento finito de condução bidimensional, considera-se uma aproximação geral inicial, ou seja, não se considera uma geometria específica de um elemento existente. Assume-se um elemento que possui M nós tal que a distribuição de temperatura no elemento é descrita por função de interpolação, mostrada na Eq.(41).

$$T(x, y) = \sum_{i=1}^M N_i(x, y) T_i = [N]\{T\} \quad (41)$$

Sendo $N_i(x, y)$ é a função de interpolação associada à temperatura nodal T_i , $[N]$ é a matriz linha de funções de interpolação e $\{T\}$ é a matriz coluna (vetor) de temperaturas nodais.

Aplicando o método de elemento finito Galerkin, as equações residuais correspondentes à Eq.(41) (Lei de Fourier) é:

$$\iint_A N_i(x, y) \left[\frac{\partial}{\partial x} \left(k_x \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k_y \frac{\partial T}{\partial y} \right) \right] dA = 0 \quad i = 1; M \quad (42)$$

Para se desenvolver as equações de elemento finito para o caso bidimensional, necessita-se realizar a manipulação matemática. Considerando as duas primeiras integrais na equação Eq.(42), tem-se:

$$\iint_A \left[\frac{\partial}{\partial x} \left(k_x \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k_y \frac{\partial T}{\partial y} \right) \right] dA = \iint_A \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} dA \quad (43)$$

O Teorema de Green-Gauss aplicado na equação diferencial que governa o problema físico mostra uma relação geral para a utilização de qualquer elemento finito. Declarando $F(x, y)$ e $G(x, y)$ como funções contínuas definidas em uma região do plano x-y (área de um elemento), então:

$$\iint_A \frac{\partial}{\partial y} \left(\beta \frac{\partial F}{\partial x} + \beta \frac{\partial G}{\partial y} \right) = \oint_S (\beta F n_x + \beta G n_y) dS - \iint_A \left(\frac{\partial F}{\partial x} \frac{\partial \beta}{\partial x} + \frac{\partial G}{\partial y} \frac{\partial \beta}{\partial y} \right) dA \quad (44)$$

Considerando $F = k_x \frac{\partial T}{\partial x}$, $G = k_y \frac{\partial T}{\partial y}$, $\beta = N_i(x, y)$, e aplicando o teorema de Green-Gauss para obter:

$$\begin{aligned} \iint_A \left[\frac{\partial}{\partial x} \left(k_x \frac{\partial T}{\partial x} \right) N_i + \frac{\partial}{\partial y} \left(k_y \frac{\partial T}{\partial y} \right) N_i \right] dA \\ = - \oint_S (q_x n_x + q_y n_y) N_i dS - \iint_A \left(\frac{\partial T}{\partial x} k_x \frac{\partial N_i}{\partial x} + \frac{\partial T}{\partial y} k_y \frac{\partial N_i}{\partial y} \right) dA \end{aligned} \quad (45)$$

A aplicação do teorema de Green-Gauss, realizada neste desenvolvimento é a contraparte bidimensional da integração por partes de uma dimensão. O resultado é que introduziu-se os termos de gradiente de borda limite como é indicado pela primeira integral no lado direito da equação Eq. (45) e assegura que a matriz de condutância seja simétrica, pela segunda integral, como é mostrado do desenvolvimento que se segue.

Voltando a relação residual de Galerkin representada na Eq.(42) e substituindo as relações desenvolvidas mediante o teorema de Green-Gauss (sendo cuidadoso em observar os sinais aritméticos), chega-se na Eq.(46).

$$\iint_A \left(k_x \frac{\partial T}{\partial x} \frac{\partial N_i}{\partial x} + k_y \frac{\partial T}{\partial y} \frac{\partial N_i}{\partial y} \right) dA = - \oint_S (q_x n_x + q_y n_y) N_i dS \quad i = 1; M \quad (46)$$

Que representa o sistema de M equações para a formulação do elemento finito bidimensional mediante o método de Galerkin. O lado esquerdo da equação inclui a distribuição de temperatura desconhecida e o lado direito é composto da função de fluxo de calor de borda limite.

Convertendo a notação matricial para facilitar a interpretação, obtém-se:

$$\iint_A \left(k_x \left[\frac{\partial T}{\partial x} \right]^T \left[\frac{\partial N}{\partial x} \right] + k_y \left[\frac{\partial T}{\partial y} \right]^T \left[\frac{\partial N}{\partial y} \right] \right) dA \{T\}^t = - \oint_S q_s n_s [N]^T dS \quad (47)$$

O qual é da forma:

$$[K^{(e)}] \{T\} = \{f_g^{(e)}\} \quad (48)$$

Relacionando as Eq. (47) e Eq.(48), é possível determinar a matriz de condução térmica, conforme a Eq.(49).

$$[K^{(e)}] = \iint_A \left(k_x \left[\frac{\partial T}{\partial x} \right]^T \left[\frac{\partial N}{\partial x} \right] + k_y \left[\frac{\partial T}{\partial y} \right]^T \left[\frac{\partial N}{\partial y} \right] \right) dA \quad (49)$$

A qual, para um elemento que tem M nós, é uma matriz $M \times M$ simétrica. Enquanto utilizar o termo matriz de condutância, o primeiro termo integral na direita da Eq.(49) representa a condução pura (similar a matriz de rigidez).

As funções de efeito calórico sobre o elemento é descrito mediante matrizes coluna (vetores), conforme a Eq.(50).

$$\{f_g^{(e)}\} = - \oint_S q_s n_s [N]^T t dS = - \oint_S q_s n_s \{N\} dS \quad (50)$$

Onde $[N]^T = \{N\}$ é a matriz coluna $M \times 1$ (vetor) das funções de interpolação.

As Equações de Eq.(48) a Eq.(50), representam a formulação geral de um elemento finito para a condução de calor bidimensional. Nota-se também, que essas equações são válidas para um elemento arbitrário que tem M nós e, por conseguinte, qualquer ordem de suas funções de interpolação (linear, quadrático, cúbico, etc).

1.8.1. CONDIÇÕES DE CONTORNO

As condições de contorno para a condução bidimensional podem ser de dois tipos.

No perímetro S_1 da superfície limite, a temperatura pode ser escrita como um valor conhecido $T_{S_1} = T$. Em um modelo de elementos finitos onde existe um subdomínio de tal tipo, cada nó de um elemento localizado em S_1 tem temperatura conhecida e as equações de equilíbrio nodal correspondentes se voltam equações de “reação”. As forças de reação são fluxo de calor nos nós localizados sobre S_1 .

O fluxo de calor, na porção S_2 da borda limite é escrita como $q_{S_2} = q$. Isso é análogo às forças nodais específicas em um problema estrutural. Todos os nós localizados na borda S_2 , seguem a Eq.(51).

$$\{f_g^{(e)}\} = - \oint_S q * n_{s2} \{N\} dS_2 \quad (51)$$

1.9. ESTADO PLANO DE TENSÃO E DEFORMAÇÃO

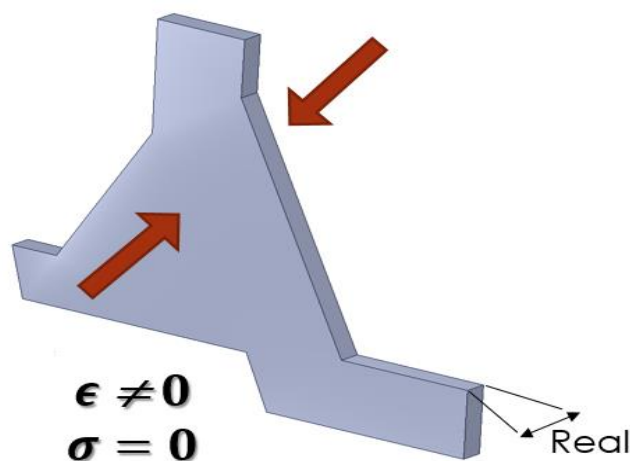
Análises de tensões podem ser resolvidas através da formulação de elemento finito para problemas de tensão e deformação. No entanto o custo computacional pode se tornar alto devido as extensas equações resultantes de um elemento tridimensional. Por isso, é muito importante a introdução de simplificações que permitam a redução do tamanho do problema sem comprometer a precisão dos resultados.

Os planos de tensões e deformações representam aproximações do comportamento de modelos tridimensionais que reduzem a complexidade do problema, tornando computacionalmente viável.

1.9.1. ESTADO PLANO DE TENSÃO

Neste caso, deve-se considerar um corpo onde uma das dimensões é muito fina quando comparada com as outras duas. Assume-se que essa espessura fina está localizada no eixo cartesiano z. O corpo então, encontra-se sobre o plano xy, conforme a **Figura 13**.

Figura 13. Sólido adequado para um modelo no estado plano de tensões



Pode-se considerar um de estado plano de tensões se as forças atuam somente no plano xy. Nesse caso, geralmente é aceitável assumir que as componentes de tensões apresentadas na Eq.(52) ,são zero.

$$\{\sigma_z, \tau_{yz}, \tau_{zx}\} = 0 \quad (52)$$

Substituindo as componentes das tensões da Eq. (52) na relação invertida de tensão e deformação, tem-se:

$$\begin{pmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{pmatrix} = \begin{pmatrix} \frac{1}{E} & -\frac{\nu}{E} & -\frac{\nu}{E} & 0 & 0 & 0 \\ -\frac{\nu}{E} & \frac{1}{E} & -\frac{\nu}{E} & 0 & 0 & 0 \\ -\frac{\nu}{E} & -\frac{\nu}{E} & \frac{1}{E} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G} \end{pmatrix} \begin{pmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \sigma_x - \nu \sigma_y \\ \sigma_y - \nu \sigma_x \\ \nu(\sigma_x - \sigma_y) \\ \tau_{xy} \\ \frac{G}{0} \\ 0 \end{pmatrix} \quad (53)$$

Reescrevendo a relação apresentada na Eq.(53), tem-se:

$$\begin{pmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{pmatrix} = \begin{pmatrix} \frac{1}{E} & -\frac{\nu}{E} & 0 \\ -\frac{\nu}{E} & \frac{1}{E} & 0 \\ 0 & 0 & \frac{1}{G} \end{pmatrix} \begin{pmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{pmatrix} \quad (54)$$

$$\varepsilon_z = \frac{\nu(\sigma_x - \sigma_y)}{E}; \quad \gamma_{zy} = 0; \quad \gamma_{zx} = 0; \quad \sigma_z = 0$$

Logo, na formulação do estado plano de tensões ε_z não é zero. No entanto, conhecendo-se σ_x e σ_y é possível determiná-lo utilizando a Eq.(54).

O problema de estado plano de tensões pode também ser reformulado em função do deslocamento ao longo de duas coordenadas direcionais u e v, seguindo as componentes de tensões e deformações, conforme a Eq.(55).

$$\text{Tensão: } \{\sigma_x, \sigma_y, \tau_{xy}\}; \quad \text{Deformação: } \{\varepsilon_x, \varepsilon_y, \gamma_{xy}\} \quad (55)$$

Invertendo a matriz, a tensão é expressa em função da deformação, conforme Eq.(56).

$$\begin{pmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{pmatrix} = \frac{E}{1-\nu^2} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{pmatrix} \begin{pmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{pmatrix} \quad (56)$$

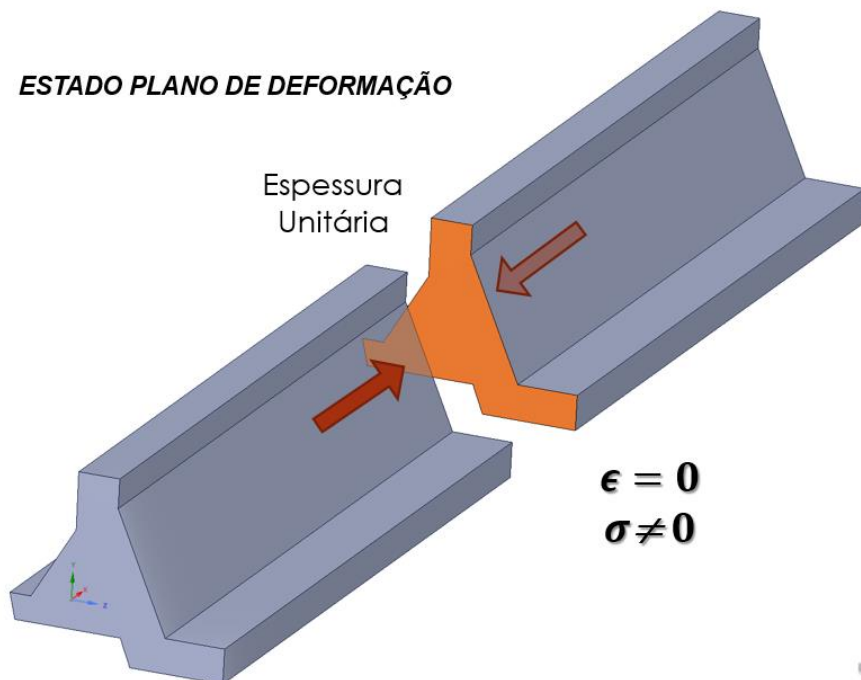
Caso haja variação de temperatura, a deformação inicial considerando o estado plano de tensões será:

$$\begin{aligned}\epsilon_0 &= (\alpha\Delta T \quad \alpha\Delta T \quad 0)^T \\ \epsilon_z &= -\frac{\nu(\sigma_x - \sigma_y)}{E} + \alpha\Delta T \\ \sigma &= C(\epsilon - \epsilon_0)\end{aligned}\quad (57)$$

1.9.2. ESTADO PLANO DE DEFORMAÇÃO

Neste caso, considera-se que uma das dimensões de um corpo é extensa quando comparada com as outras duas. Assume-se que essa direção longa está localizada no eixo cartesiano z. Considera-se então, uma fatia de espessura unitária do corpo que se encontra no plano xy, conforme a **Figura 14**. As forças aplicadas no corpo, assim como no caso do estado plano de tensão, estão localizadas no eixo xy. Assume-se que a deformação no eixo z é zero. Pode-se considerar então, que o corpo se encontra no estado plano de deformação.

Figura 14. Sólido adequado para um modelo no estado plano de deformação.



Fonte: ASGHAR (2005),

Assumindo que ϵ_z , γ_{yz} e γ_{zx} são iguais a zero, a relação entre tensão e deformação fica conforme a Eq.(58).

$$\begin{pmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{pmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-\nu}{2} \end{pmatrix} \begin{pmatrix} \epsilon_x \\ \epsilon_y \\ 0 \\ \gamma_{xy} \\ 0 \\ 0 \end{pmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} (1-\nu)\epsilon_x - \nu\epsilon_y \\ (1-\nu)\epsilon_y - \nu\epsilon_x \\ \nu(\epsilon_x - \epsilon_y) \\ (1-2\nu)\gamma_{xy} \\ 0 \\ 0 \end{pmatrix} \quad (58)$$

Reescrevendo a Eq.(58), tem-se:

$$\begin{pmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{pmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{pmatrix} \begin{pmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{pmatrix} = \frac{E\nu(\epsilon_x + \epsilon_y)}{(1+\nu)(1-2\nu)} \quad (59)$$

$$\tau_{xy} = 0; \quad \tau_{zx} = 0$$

De acordo com Eq.(59), a tensão normal em z (σ_z) é diferente de zero na formulação do estado plano de deformação.

Semelhante a formulação do estado plano de tensão, o estado plano de deformação pode ser expresso em termos do deslocamento ao longo de duas direções de coordenadas u e v. Efetivamente, a única diferença entre o estado plano de tensão e o de deformação é a matriz constitutiva.

Se houver variação de temperatura, a deformação inicial para o estado plano de deformação é estabelecida conforme a Eq.(60).

$$\sigma = C(\epsilon - \epsilon_0)$$

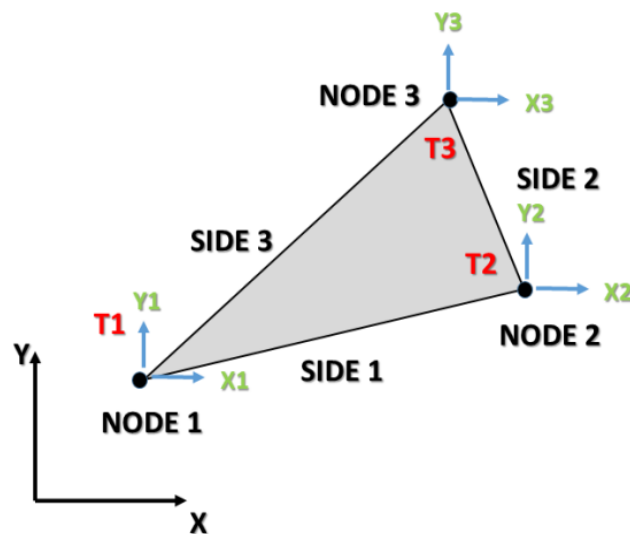
$$\epsilon_0 = (1 + \nu)(\alpha\Delta T \quad \alpha\Delta T \quad 0)^T$$

$$\begin{pmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{pmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{pmatrix} \begin{pmatrix} \epsilon_x - (1+\nu)\alpha\Delta T \\ \epsilon_y - (1+\nu)\alpha\Delta T \\ \gamma_{xy} \end{pmatrix} = \frac{E\nu(\epsilon_x + \epsilon_y)\alpha\Delta T}{(1+\nu)(1-2\nu)} \quad (60)$$

1.10. ELEMENTO FINITO TRIANGULAR DE TRÊS NÓS

De acordo com ASGHAR (2005), um elemento triangular é simples e versátil para a resolução de problemas bidimensionais. Quase todas as formas podem ser discretizadas utilizando o elemento triangular. Na **Figura 15** é apresentado um elemento triangular.

Figura 15. Elemento Triangular de Três Nós



Fonte: ASGHAR (2005),

As coordenadas nodais dos nós 1, 2 e 3 são (x_1, y_1) , (x_2, y_2) e (x_3, y_3) , respectivamente. A direção normal de cada lado do elemento é representada pelo vetor n . A coordenada de origem de cada contorno é representada pelo vetor c , que também representa o primeiro nó de cada lado. Assim, a variável c pode assumir valores que vão de zero (0), até o comprimento da aresta de cada lado do elemento.

O erro associado a discretização de uma curva de contorno através de várias linhas retas representadas pelas arestas do elemento, pode ser reduzido aumentando o número de elementos nessa região. No caso do problema de campo de temperatura, cada nó possui um grau de liberdade (gdl), a temperatura. A solução do elemento terá a forma apresentada na **Eq.(61)**.

Na área do elemento:

$$u(x, y) = (N_1(x, y) \ N_2(x, y) \ N_3(x, y)) \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = N^T d \quad (61)$$

No contorno do elemento:

$$u(c) = (N_1(c) \ N_2(c) \ N_3(c)) \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = N_c^T d$$

Um polinômio linear completo em duas dimensões tem três termos e a solução do elemento finito é baseado no seguinte polinômio:

$$u(x, y) = c_0 + c_1x + c_2y = (1 \ x \ y) \begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} \quad (62)$$

Os coeficientes c_0 , c_1 e c_2 expressos em termos dos graus de liberdade nodais são descritos na Eq.(63).

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} \quad (63)$$

Invertendo a matriz 3x3, tem-se:

$$\begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} = \frac{1}{2A} \begin{pmatrix} f_1 & f_2 & f_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \quad (64)$$

Sendo A = Área do elemento triangular;

$$\begin{aligned} f_1 &= x_2y_3 - x_3y_2; & f_2 &= x_3y_1 - x_1y_3; & f_3 &= x_1y_2 - x_2y_1; \\ b_1 &= y_2 - y_3; & b_2 &= y_3 - y_1; & b_3 &= y_1 - y_2; \\ c_1 &= x_3 - x_2; & c_2 &= x_1 - x_3; & c_3 &= x_2 - x_1; \end{aligned}$$

Substituindo os coeficientes na equação polinomial, tem-se:

$$u(x, y) = (1 \ x \ y) \frac{1}{2A} \begin{pmatrix} f_1 & f_2 & f_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \quad (65)$$

Realizando as multiplicações, obtém-se a solução assumida do elemento finito, conforme a Eq.(66).

$$u(x, y) = (N_1 \ N_2 \ N_3) \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = N^T d \quad (66)$$

Sendo:

$$\begin{aligned} N_1 &= \frac{1}{2A} (x_3(y - y_2) + x(y_2 - y_3) + x_2(-y + y_3)) \equiv \frac{1}{2A} (xb_1 + yc_1 + f_1) \\ N_2 &= \frac{1}{2A} (x_3(-y + y_1) + x_1(y - y_3) + x(-y_1 + y_3)) \equiv \frac{1}{2A} (xb_2 + yc_2 + f_2) \\ N_3 &= \frac{1}{2A} (x_2(y - y_1) + x(y_1 - y_2) + x_1(-y + y_2)) \equiv \frac{1}{2A} (xb_3 + yc_3 + f_3) \end{aligned}$$

A borda é definida pelos três lados do elemento. Para cada lado a solução é linear em cada lado em termos da coordenada dos dois graus de liberdade localizados nas extremidades de cada aresta do elemento. Utilizando a fórmula de interpolação de Lagrange é possível obter a solução linear, conforme a Eq.(67).

Para o Lado 1

$$u(c) = \left(-\frac{c - L_{12}}{L_{12}} \frac{c}{L_{12}} \ 0 \right) \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix}; \quad 0 \leq c \leq L_{12}$$

$$\text{Sendo } L_{12} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (\text{comprimento do lado 1-2}).$$

Para o Lado 2

$$u(c) = \left(0 \ -\frac{c - L_{23}}{L_{23}} \frac{c}{L_{23}} \right) \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix}; \quad 0 \leq c \leq L_{23} \quad (67)$$

$$\text{Sendo } L_{23} = \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2} \quad (\text{comprimento do lado 2-3}).$$

Para o Lado 3

$$u(c) = \left(\frac{c}{L_{31}} \ 0 \ -\frac{c - L_{31}}{L_{31}} \right) \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix}; \quad 0 \leq c \leq L_{31}$$

$$\text{Sendo } L_{23} = \sqrt{(x_3 - x_1)^2 + (y_3 - y_1)^2} \quad (\text{comprimento do lado 3-1}).$$

Resolvendo a integral no elemento, as matrizes e vetores necessários para a equação do elemento pode ser escrito conforme a Eq.(68).

$$k_k = \iint_A BCB^T dA; \quad k_\alpha = - \int_{c_n} \alpha N_c N_c^T dc; \quad r_q = \iint_A qNdA; \quad r_\beta = \int_{c_n} \beta N_c dc; \quad (68)$$

1.10.1. PROBLEMA TÉRMICO (REGIME PERMANENTE)

No caso do problema de campo de temperatura, cada nó possui um grau de liberdade, a temperatura. Considerando uma temperatura de borda como condição de contorno e que o modelo não possui fonte de calor, a equação do elemento, assume a forma apresentada na Eq.(69).

$$k_k T = 0 \quad (69)$$

Sendo k_k a matriz de rigidez, dada pela Eq. (70).

$$k_k = \iint_A BCB^T dA = ABCB^T \quad (70)$$

e

$$C = \begin{pmatrix} k_x & 0 \\ 0 & k_y \end{pmatrix} \quad (71)$$

Considerando um elemento triangular de três nós, B^T fica conforme a Eq.(72).

$$B^T = \begin{pmatrix} \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial x} & \frac{\partial N_3}{\partial x} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_2}{\partial y} & \frac{\partial N_3}{\partial y} \end{pmatrix} = \frac{1}{2A} \begin{pmatrix} b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{pmatrix} \quad (72)$$

E a matriz fica conforme a Eq.(73).

$$k_k = \frac{1}{4A} \begin{pmatrix} k_x b_1^2 + k_y c_1^2 & k_x b_1 b_2 + k_y c_1 c_2 & k_x b_1 b_3 + k_y c_1 c_3 \\ k_x b_1 b_2 + k_y c_1 c_2 & k_x b_2^2 + k_y c_2^2 & k_x b_2 b_3 + k_y c_2 c_3 \\ k_x b_1 b_3 + k_y c_1 c_3 & k_x b_2 b_3 + k_y c_2 c_3 & k_x b_3^2 + k_y c_3^2 \end{pmatrix} \quad (73)$$

Sendo b_1 , b_2 e b_3 variáveis relacionadas com as coordenadas nodais do elemento.

k_x e k_y são os coeficientes de condutividade térmica do material na direção horizontal e vertical, respectivamente.

1.10.2. PROBLEMA TÉRMO-MECÂNICO

Inicialmente, tanto para o estado plano de tensão como o de deformação a formulação das componentes de deslocamento no u e v . A única diferença nas duas formulações são as equações constitutivas. Assim, ambas as formulações podem ser elaboradas essencialmente na mesma formulação de elementos finitos. No caso do estado plano de tensão, deve-se adotar uma espessura h do modelo. No caso do estado plano de deformação, a espessura será unitária.

A formulação das equações do elemento finito para o problema Termo-Estrutural, considerando somente a deformação inicial devido a variação de temperatura é apresentado na Eq.(74).

$$kd = r_q + r_b + r_\epsilon \quad (74)$$

Sendo d o vetor de deslocamentos nodais, r_q e r_b são vetores de carregamentos externos e k a matriz de rigidez, apresentada na Eq.(75).

$$k = \iint_A BCB^T dA \quad (75)$$

Sendo B a matriz de transformação e C a matriz constitutiva, apresentadas nas Eq.(76) e (77).

$$B^T = \frac{1}{2A} \begin{pmatrix} b_1 & 0 & b_2 & 0 & b_3 & 0 \\ 0 & c_1 & 0 & c_2 & 0 & c_3 \\ c_1 & b_1 & c_2 & b_2 & c_3 & b_3 \end{pmatrix} \quad (76)$$

Para o estado plano de tensão:

$$C = \frac{E}{(1-\nu^2)} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{pmatrix} \quad (77)$$

Para o estado plano de deformação:

$$C = \frac{E}{(1+\nu)(1+2\nu)} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{pmatrix}$$

Trabalhando na **Eq.(77)**, chega-se na matriz de rigidez do problema mecânico bidimensional, para o estado plano de tensão e deformação, conforme **Eq.(78)**.

Se considerar o Estado plano de Tensão, então:

$$const 1 = \frac{E}{1 - \nu^2}; \quad const 2 = \frac{E\nu}{1 - \nu^2}; \quad const 3 = \frac{E(1 - \nu)}{2(1 - \nu^2)}$$

Caso contrário, se considerar o Estado Plano de Deformação, então:

$$const 1 = \frac{E(1 - \nu)}{(1 + \nu)(1 - 2\nu)}; \quad const 2 = \frac{E\nu}{(1 + \nu)(1 - 2\nu)}; \quad const 3 = \frac{E(1 - 2\nu)}{2(1 + \nu)(1 - 2\nu)}$$

$k =$

$$\begin{pmatrix} \left(\frac{b_1}{2A}\right)^2 const 1 + \left(\frac{c_1}{2A}\right)^2 const 3 & \frac{b_1 c_1}{4A^2} (const 2 + const 3) & \frac{b_2 b_1}{4A^2} const 1 + \frac{c_1 c_2}{4A^2} const 3 & \frac{c_2 b_1}{4A^2} const 2 + \frac{b_2 c_1}{4A^2} const 3 & \frac{b_3 b_1}{4A^2} const 1 + \frac{c_3 c_1}{4A^2} const 3 & \frac{c_3 b_1}{4A^2} const 1 + \frac{b_3 c_1}{4A^2} const 3 \\ \frac{b_1 c_1}{4A^2} (const 2 + const 3) & \left(\frac{c_1}{2A}\right)^2 const 1 + \left(\frac{b_1}{2A}\right)^2 const 3 & \frac{b_2 c_1}{4A^2} const 2 + \frac{c_2 b_1}{4A^2} const 3 & \frac{c_2 c_1}{4A^2} const 1 + \frac{b_2 b_1}{4A^2} const 3 & \frac{b_3 c_1}{4A^2} const 2 + \frac{c_3 b_1}{4A^2} const 3 & \frac{c_3 c_1}{4A^2} const 1 + \frac{b_3 b_1}{4A^2} const 3 \\ \frac{b_2 b_1}{4A^2} const 1 + \frac{c_1 c_2}{4A^2} const 3 & \frac{c_1 b_2}{4A^2} const 2 + \frac{b_1 c_2}{4A^2} const 3 & \left(\frac{b_1}{2A}\right)^2 const 1 + \left(\frac{c_2}{2A}\right)^2 const 3 & \frac{b_2 c_2}{4A^2} (const 2 + const 3) & \frac{b_2 b_3}{4A^2} const 1 + \frac{c_2 c_3}{4A^2} const 3 & \frac{b_2 c_3}{4A^2} const 2 + \frac{b_3 c_2}{4A^2} const 3 \\ \frac{c_2 b_1}{4A^2} const 2 + \frac{c_1 b_2}{4A^2} const 3 & \frac{c_1 c_2}{4A^2} const 1 + \frac{b_1 b_2}{4A^2} const 3 & \frac{b_2 c_2}{4A^2} (const 2 + const 3) & \left(\frac{c_2}{2A}\right)^2 const 1 + \left(\frac{b_2}{2A}\right)^2 const 3 & \frac{b_3 c_2}{4A^2} const 2 + \frac{c_3 b_2}{4A^2} const 3 & \frac{c_2 c_3}{4A^2} const 1 + \frac{b_2 b_3}{4A^2} const 3 \\ \frac{b_3 b_1}{4A^2} const 1 + \frac{c_1 c_3}{4A^2} const 3 & \frac{b_3 c_1}{4A^2} const 2 + \frac{b_1 c_3}{4A^2} const 3 & \frac{b_2 b_3}{4A^2} const 1 + \frac{c_2 c_3}{4A^2} const 3 & \frac{b_2 c_3}{4A^2} const 2 + \frac{b_3 c_2}{4A^2} const 3 & \left(\frac{b_1}{2A}\right)^2 const 1 + \left(\frac{c_3}{2A}\right)^2 const 3 & \frac{b_3 c_3}{4A^2} (const 2 + const 3) \\ \frac{b_2 c_3}{4A^2} const 2 + \frac{b_3 c_2}{4A^2} const 3 & \frac{c_1 c_3}{4A^2} const 1 + \frac{b_1 b_3}{4A^2} const 3 & \frac{b_2 c_3}{4A^2} const 2 + \frac{b_3 c_2}{4A^2} const 3 & \frac{c_2 c_3}{4A^2} const 1 + \frac{b_2 b_3}{4A^2} const 3 & \frac{b_3 c_3}{4A^2} (const 2 + const 3) & \left(\frac{c_3}{2A}\right)^2 const 1 + \left(\frac{b_3}{2A}\right)^2 const 3 \end{pmatrix} \quad (78)$$

e r_ϵ é o vetor de carregamento devido as deformações iniciais originadas na variação de temperatura do corpo, conforme a Eq.(79).

$$r_\epsilon = h \iint_A BC\epsilon_0 dA \quad (79)$$

Sendo

Para o estado plano de tensão: $\epsilon_0 = (\alpha\Delta T \quad \alpha\Delta T \quad 0)^T$

Para o estado plano de deformação: $\epsilon_0 = (1 + \nu)(\alpha\Delta T \quad \alpha\Delta T \quad 0)^T$

1.10.3. ACOPLAMENTO DO PROBLEMA TÉRMO-MECÂNICO

O acoplamento entre o problema térmico e o mecânico é baseado na Eq. (79).

Declara-se um valor de temperatura ambiente padrão, no qual o material não apresenta deformações térmicas (Standard Temperature). Em seguida, subtrai-se a temperatura padrão das temperaturas nodais resultante do problema térmico. O resultado é um vetor com a variação de temperatura nodal (ΔT), conforme a Eq.(80).

$$\Delta T_i = T_i - T_{padr\tilde{a}o} \quad (80)$$

Sendo ΔT a variação de temperatura, T_i as temperaturas de cada nó.

Posteriormente, multiplica-se o vetor ΔT_i pelo coeficiente de dilatação térmica do material (α). O resultado é o vetor de deformação inicial devido a variação de temperatura de cada nó, conforme a Eq.(81).

$$\epsilon_{0_i} = \alpha \Delta T_i = \alpha \left(\begin{array}{c} T_{n\acute{o} 1} \\ T_{n\acute{o} 2} \\ T_{n\acute{o} 3} \\ \vdots \\ T_{n\acute{o} n} \end{array} \right) - (\text{Temperatura Padr\~{a}o}) \quad (81)$$

Sendo ϵ_{0_i} o vetor de deformação térmica, α o coeficiente de dilatação térmica.

Baseado na Equação constitutiva da elasticidade, Eq.(39), é possível deduzir as forças interna devido a deformação térmica. No caso do MEF, o vetor de forças nodais do elemento deduzido a partir da variação de temperatura é apresentado na Eq.(82).

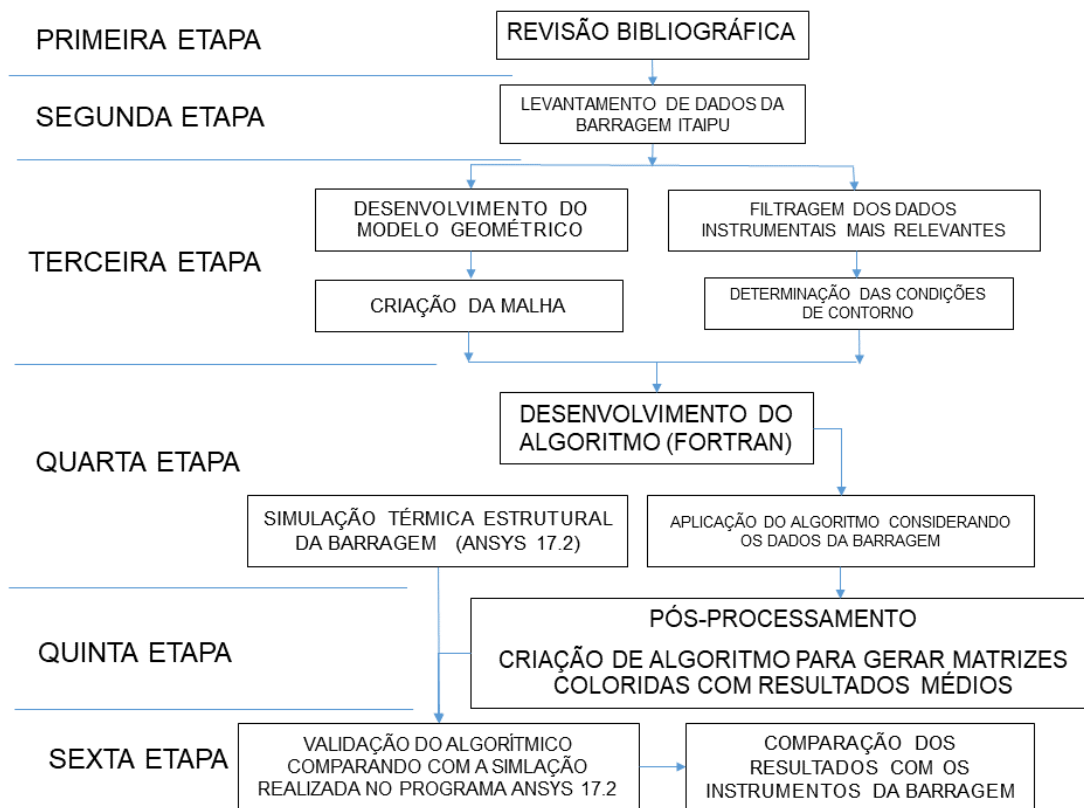
$$r_\epsilon = hABC\epsilon_0 \quad (82)$$

Sendo r_ϵ o vetor de forças nodais, h a espessura do elemento, B a matriz de transformação, C a matriz constitutiva da elasticidade.

2. METODOLOGIA

Neste capítulo é apresentado os métodos que foram adotados para se alcançar os objetivos definidos anteriormente. O trabalho foi dividido em 6 (seis) etapas, conforme a **Figura 16**.

Figura 16. Metodologia adotada.



PRIMEIRA ETAPA - REVISÃO BIBLIOGRÁFICA

Consiste na elaboração da revisão bibliográfica e fundamentação teórica apresentados nos capítulos 1 e 2, respectivamente. Essa etapa contribuiu para o embasamento teórico através dos trabalhos de outros autores que estudam o problema da condução de calor aplicado em barragens e por meio das ideias captadas dos autores conseguiu-se adotar a seguinte metodologia, dividida desde a segunda (2ª) até a quinta (5ª) etapa.

SEGUNDA ETAPA - LEVANTAMENTO DE DADOS

Na segunda etapa, realizou-se o levantamento de dados do Bloco E-6 da barragem de ITAIPU. Entre os principais dados estão: medidas do perfil da

barragem, valores de temperatura dos termômetros e os valores de deslocamento do pêndulo direto.

TERCEIRA ETAPA - PRÉ-PROCESSAMENTO

Na terceira etapa desenvolveu-se um modelo geométrico simplificado do Bloco E-6. Determinou-se as condições de contorno de acordo com a localização dos termômetros. Criou-se a malha através da divisão do domínio em várias partes menores, considerando um elemento triangular.

QUARTA ETAPA - PROCESSAMENTO

Na quarta etapa foi desenvolvido um algoritmo utilizando a linguagem de programação FORTRAN, capaz de calcular a transferência de calor em um meio bidimensional considerando regime permanente. Posteriormente, o código automaticamente utiliza o campo de temperatura resultante como condição de deformação inicial para o problema mecânico, através da variação de temperatura.

QUINTA ETAPA – PÓS-PROCESSAMENTO

Finalmente, na quinta etapa, os resultados são processados automaticamente pelo algoritmo através de uma rotina que cria uma matriz de acordo com número de elementos do problema e aloca os valores de temperatura, deslocamento e tensões em cada elemento da matriz, permitindo uma visualização rápida e intuitiva dos resultados de forma gráfica com cores.

SEXTA ETAPA – COMPARAÇÃO E VALIDAÇÃO

Nesta etapa foi realizado simulações térmicas-estruturais do Bloco E6 da CHI, utilizando o programa ANSYS® 17.2. Essas simulações foram utilizadas para comparar e validar o algoritmo desenvolvido neste trabalho. Posteriormente, comparou-se os resultados dos modelos com os instrumentos da barragem para se compreender o comportamento térmico de uma estrutura de concreto massa.

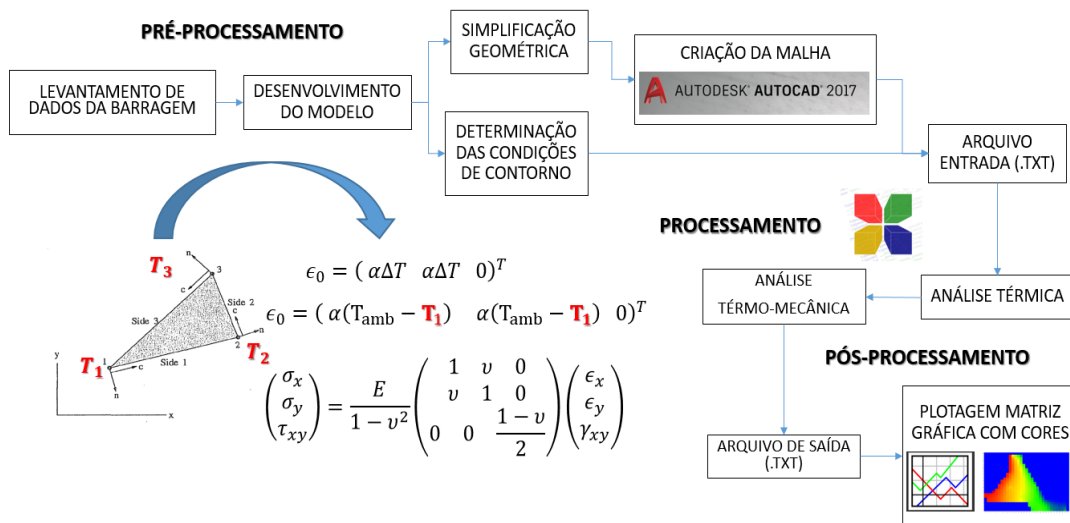
3. ALGORITMO EM FORTRAN

O algoritmo foi desenvolvido na linguagem de programação *FORTRAN 95* através do programa *CODEBLOCKS*, uma plataforma livre de programação.

As informações da malha, propriedades dos materiais, condições de contorno, entre outras, são alocadas em variáveis do programa através de um arquivo de texto (.txt) de entrada. Após o processamento, os resultados são armazenados em forma de lista, em vetores, e em campos de valores, em matrizes. Os resultados matriciais são armazenados em arquivos de texto (.txt) de saída e podem ser plotados graficamente com o auxílio do programa *GNU PLOT*, um pacote livre de múltiplas plataformas gráficas que funciona através de linhas de comando. Esse programa converte os valores das matrizes em cores, o que permite comparar os resultados do algoritmo e do programa Ansys®.

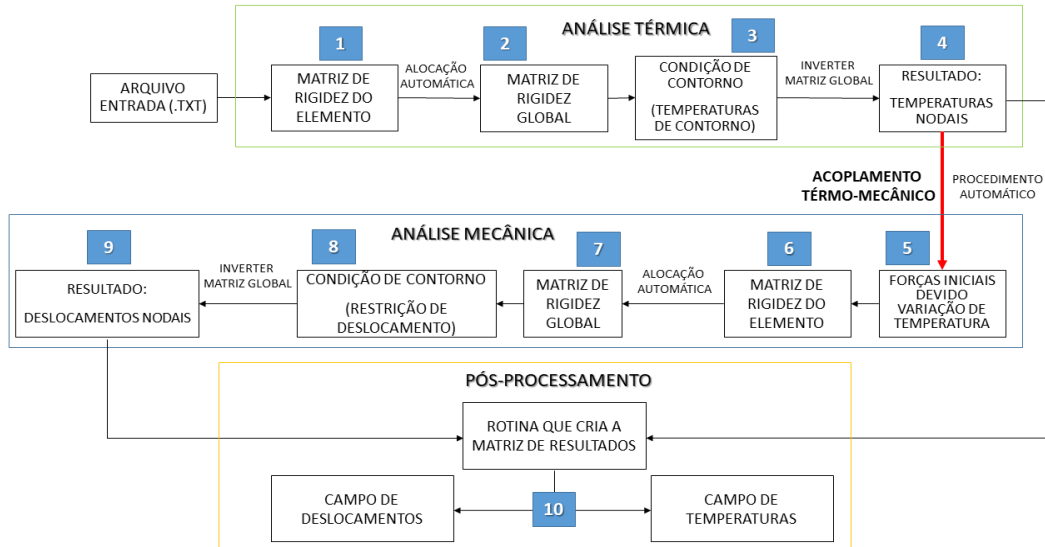
Na **Figura 17** é apresentado o diagrama esquemático geral da resolução do problema e os programas que foram utilizados em cada fase.

Figura 17. Diagrama Esquemático Geral da resolução do problema



O algoritmo pode ser dividido em 10 partes para ilustrar o seu funcionamento, conforme ilustra a **Figura 18**.

Figura 18. Diagrama do algoritmo



Uma breve explicação sobre cada parte do algoritmo é apresentada nos subcapítulos subsequentes.

3.1. MATRIZ DE RIGIDEZ DO ELEMENTO (TÉRMICO)

O processamento do problema térmico se inicia na resolução da matriz de rigidez do elemento. O algoritmo apresentado na **PARTE 1**, utilizando a linguagem de programação FORTRAN, foi baseado nas **Eq. (64) e (73)**.

Parte 1. Algoritmo da matriz de rigidez do elemento triangular(análise térmica)

```

|*****
! COMPONENTES DA MATRIZ DE RIGIDEZ DO ELEMENTO
|*****
allocate(F1(ELEM_NUMBER),F2(ELEM_NUMBER),F3(ELEM_NUMBER)); F1=0.D0;F2=0.D0;F3=0.D0
allocate(B1(ELEM_NUMBER),B2(ELEM_NUMBER),B3(ELEM_NUMBER)); B1=0.D0;B2=0.D0;B3=0.D0
allocate(C1(ELEM_NUMBER),C2(ELEM_NUMBER),C3(ELEM_NUMBER)); C1=0.D0;C2=0.D0;C3=0.D0
allocate(ELEMENT_AREA(ELEM_NUMBER)); ELEMENT_AREA=0.D0

allocate(L_CONDUCTION_MATRIX(ELEM_NUMBER,3,3));L_THERMAL_MATRIX=0.D0
do i=1, ELEM_NUMBER

    X1=NODE_POSITION(ELEM_NODE(i,1),1)
    X2=NODE_POSITION(ELEM_NODE(i,2),1)
    X3=NODE_POSITION(ELEM_NODE(i,3),1)

    Y1=NODE_POSITION(ELEM_NODE(i,1),2)
    Y2=NODE_POSITION(ELEM_NODE(i,2),2)
    Y3=NODE_POSITION(ELEM_NODE(i,3),2)

    F1(i)= X2*Y3-X3*Y2
    
```

```

F2(i)= X3*Y1-X1*Y3
F3(i)= X1*Y2-X2*Y1

B1(i)= Y2-Y3
B2(i)= Y3-Y1
B3(i)= Y1-Y2

C1(i)= X3-X2
C2(i)= X1-X3
C3(i)= X2-X1

ELEMENT_AREA(i) = (F1(i)+F2(i)+F3(i))/2

|*****
! MATRIZ DE RIGIDEZ DO ELEMENTO
|*****
AREA=ELEMENT_AREA(i)
k_coef=MATERIAL_PROP(ELEM_MATERIAL(i),3)

L_CONDUCTION_MATRIX(i,1,1)= k_coef*(B1(i)**2+C1(i)**2)*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,1,2)= k_coef*(B1(i)*B2(i)+C1(i)*C2(i))*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,1,3)= k_coef*(B1(i)*B3(i)+C1(i)*C3(i))*(1/(4*AREA))

L_CONDUCTION_MATRIX(i,2,1)= k_coef*(B1(i)*B2(i)+C1(i)*C2(i))*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,2,2)= k_coef*(B2(i)**2+C2(i)**2)*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,2,3)= k_coef*(B2(i)*B3(i)+C2(i)*C3(i))*(1/(4*AREA))

L_CONDUCTION_MATRIX(i,3,1)= k_coef*(B1(i)*B3(i)+C1(i)*C3(i))*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,3,2)= k_coef*(B2(i)*B3(i)+C2(i)*C3(i))*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,3,3)= k_coef*(B3(i)**2+C3(i)**2)*(1/(4*AREA))

end do

```

3.2. MATRIZ GLOBAL DE RIGIDEZ (TÉRMICO)

Uma das coisas mais importantes na programação do MEF é a determinação da matriz global de rigidez de forma automática a partir das matrizes de rigidez de cada elemento. Considerando um elemento triangular linear, cada elemento possui três nós. No caso do problema térmico, cada nó possui apenas um grau de liberdade. Portanto, a matriz de rigidez global pode ser formada automaticamente considerando a **PARTE 2** do algoritmo.

Parte 2. Algoritmo da matriz de rigidez global (análise térmica)

```

|*****
|*****MATRIZ GLOBAL DE RIGIDEZ *****
|*****
allocate(G_THERMAL_MATRIX(NODE_NUMBER,NODE_NUMBER), V_GLOBAL(NODE_NUMBER));
G_THERMAL_MATRIX=0.D0; V_GLOBAL=0.D0

do l=1, ELEM_NUMBER
do j=1,3
do k=1, 3

G_THERMAL_MATRIX(ELEM_NODE(i,j),ELEM_NODE(i,k))=G_THERMAL_MATRIX(ELEM_NODE(i,j),ELEM_NODE(i,k))+L_CONDUCTION_MATRIX(i,j,k)

end do
end do
END do

```

3.3. CONDIÇÕES DE CONTORNO

Após a matriz global de rigidez, aplica-se as condições de contorno. No caso do problema de transferência de calor considerando unicamente temperaturas impostas como condição de contorno, deve-se realizar algumas operações que envolvem a matriz de rigidez global, o vetor de temperaturas, as temperaturas de contorno e o vetor de fluxo. Essa rotina é apresentada na **PARTE 3**.

Parte 3. Implementação das Condições de contorno (análise térmica)

```

|*****
! APLICAR TEMPERATURAS DE CONTORNO NO VETOR GLOBAL DE TEMPERATURAS
|*****
ALLOCATE(VETOR_TEMP_NODE(NODE_NUMBER));VETOR_TEMP_NODE=0.D0

do i=1, NODE_NUMBER
  if(ID_NODE_NBC(i,4)==1)then
    VETOR_TEMP_NODE(i)=NBC_TEMP(i)
  endif
end do

|*****
! DELETAR TODAS AS LINHAS QUE REPRESENTAM OS NÓS QUE POSSUEM NBC DE TEMPERATURA
|*****
do i=1, NODE_NUMBER
  IF(ID_NODE_NBC(i,4)==1)THEN
    G_THERMAL_MATRIX(i,:)=0.D0
    G_THERMAL_MATRIX(i,i)=1.D0
  END IF
END DO

|*****
! MULTIPLICAR AS COLUNAS DOS NÓS COM NBC DE TEMPERATURA PELO VETOR NBC_TEMP
|*****
allocate(VETOR_TEMP_NODE_mult(NODE_NUMBER));VETOR_TEMP_NODE_mult=1.D0
allocate(VETOR_PIVOT(NODE_NUMBER));VETOR_PIVOT=0.D0
do i=1, NODE_NUMBER
  if(ID_NODE_NBC(i,4)==1)then
    VETOR_TEMP_NODE_mult(i)=VETOR_TEMP_NODE(i)
  end if
end do

|*****
! SOMAR O (SOMATÓRIO DAS COLUNAS * NBC TEMP) DE CADA NÓ COM O VETOR DE FLUXO. INVERTER
SINAL!!!
|*****
do i=1, NODE_NUMBER
do j=1, NODE_NUMBER
  IF(ID_NODE_NBC(i,4)==1)THEN
    VETOR_PIVOT(j)= VETOR_PIVOT(j)+G_THERMAL_MATRIX(j,i)*VETOR_TEMP_NODE_mult(i)
  END IF
END DO
END DO

V_GLOBAL=V_GLOBAL+(-vetor_PIVOT) !AQUI INVERTE SINAL

do i=1, NODE_NUMBER
  if(ID_NODE_NBC(i,4)==1)then
    V_GLOBAL(i)=0.D0
  endif
end do

```

3.4. RESULTADO – TEMPERATURAS NODAIS

Finalmente, o campo de temperatura é obtido a após a inversão da matriz global de rigidez. Para isso, utilizou-se duas sub-rotinas, **ludcmp** e **lubksb**, apresentadas no **ANEXO I**. As sub-rotinas são alocadas no final do código e chamadas para inverter a matriz, conforme é apresentado na **PARTE 4**.

Parte 4. Inversão da matriz global e obtenção do vetor de temperatura (análise térmica)

```

|*****
!SUBROTINAS QUE RESOLVEM O SISTEMA LU
|*****
allocate (THERMAL_INDX(NODE_NUMBER))

call ludcmp(G_THERMAL_MATRIX,NODE_NUMBER,NODE_NUMBER,THERMAL_INDX,ddd)
call lubksb(G_THERMAL_MATRIX,NODE_NUMBER,NODE_NUMBER,THERMAL_INDX,V_GLOBAL)

|*****
! RESULTADO DAS TEMPERATURAS NODAIS
|*****
VETOR_TEMP_NODE=VETOR_TEMP_NODE+V_GLOBAL

!WRITE(*,*)"*****"
! WRITE(*,*)"RESULTADO:"
! WRITE(*,*)"NODE, TEMPERATURE"
! do i=1,NODE_NUMBER
! WRITE(*,*) i, VETOR_TEMP_NODE(i)
! end do

```

Finalmente, as temperaturas de cada nó são alocadas no vetor **VETOR_TEMP_NODE**. Cada posição do vetor corresponde a um nó da malha.

3.5. ACOPLAMENTO TÉRMO-MECÂNICO

A implementação do vetor de força iniciais responsável pelo acoplamento termo-mecânico no código é apresentada no **PARTE 5**.

Parte 5. Rotina de acoplamento entre o problema térmico e o mecânico

```

|*****
! CONTRIBUIÇÃO DA TEMPERATURA --> The equivalent load vector from initial strains due to temperature change is
|*****
! AS TEMPERATURAS NODAIS SÃO UTILIZADAS COMO CONDIÇÃO DE FORÇA INICIAL DEVIDO A DILATAÇÃO TÉRMICA.
! VARIÁVEL DA TEMPERATURA NODAL VETOR_TEMP_NODE(ELEM_NODE(i,j))

do i=1, ELEMENT_NUMBER

L_re_VETOR(i,1)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,1))-STANDARD_TEMP)*B1(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))
L_re_VETOR(i,2)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,1))-STANDARD_TEMP)*C1(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))
L_re_VETOR(i,3)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,2))-STANDARD_TEMP)*B2(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))
L_re_VETOR(i,4)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,2))-STANDARD_TEMP)*C2(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))
L_re_VETOR(i,5)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,3))-STANDARD_TEMP)*B3(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))
L_re_VETOR(i,6)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,3))-STANDARD_TEMP)*C3(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))

|*****
end do

```

3.6. MATRIZ DE RIGIDEZ DO ELEMENTO (MECÂNICA)

Após a conclusão do problema térmico, inicia-se o processamento do problema termo-mecânico. A matriz de rigidez do problema mecânico é apresentada na **PARTE 6**. Nesse caso, cada nó possui dois graus de liberdade, deslocamento na horizontal e na vertical. Portanto, a matriz de rigidez do elemento, utilizando um elemento triangular linear, será uma matriz 6x6. Pode-se considerar o estado plano de tensão ou deformação, de acordo com a hipótese adotada. A rotina foi baseada nas Eq.(75) e (76).

Parte 6. Matriz local do elemento (análise Mecânica)

```

|*****
! COMPONENTES DA MATRIZ DE RIGIDEZ DO ELEMENTO
|*****
allocate(LOCAL_TS_MATRIX(ELEM_NUMBER,6,6));LOCAL_TS_MATRIX=0.DO
allocate(L_re_VETOR(ELEM_NUMBER,6));L_re_VETOR=0.DO

do i=1, ELEM_NUMBER

MOD_ELAST=MATERIAL_PROP(ELEM_MATERIAL(i),1)
POISSON=MATERIAL_PROP(ELEM_MATERIAL(i),2)

|*****
! MATRIZ DE RIGIDEZ DO ELEMENTO
|*****
AREA=ELEMENT_AREA(i)

if((ID_STRESS_STRAIN(i)==1)then

|*****
! ESTADO PLANO DE TENSÃO !!!!!!!
|*****
CONST1=(MOD_ELAST/(1-POISSON**2))
CONST2=(MOD_ELAST*POISSON/(1-POISSON**2))
CONST3= (MOD_ELAST*(1-POISSON)/(2*(1-POISSON**2)))
DIV_CONST1=(4*AREA**2)
hA=ELEM_THICKNESS(i)*ELEMENT_AREA(i)

CONST4 = MATERIAL_PROP(ELEM_MATERIAL(i),4)

Else

|*****
! ESTADO PLANO DE DEFORMAÇÃO !!!!!!!
|*****
CONST1=(MOD_ELAST*(1-POISSON)/((1+POISSON)*(1-2*POISSON)))
CONST2=(MOD_ELAST*POISSON/((1+POISSON)*(1-2*POISSON)))
CONST3= (MOD_ELAST*(1-2*POISSON)/(2*(1+POISSON)*(1-2*POISSON)))
DIV_CONST1=(4*AREA**2)
hA=ELEM_THICKNESS(i)*ELEMENT_AREA(i)
CONST4 = (1+POISSON)*MATERIAL_PROP(ELEM_MATERIAL(i),4)
end if

|*****
LOCAL_TS_MATRIX(i,1,1)= (B1(i)**2*CONST1/DIV_CONST1 + C1(i)**2*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,1,2)= ((C1(i)*B1(i))/DIV_CONST1*(CONST2 + CONST3))*hA
LOCAL_TS_MATRIX(i,1,3)= (B2(i)*B1(i)*CONST1/DIV_CONST1 + C2(i)*C1(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,1,4)= (B1(i)*C2(i)*CONST2/DIV_CONST1 + B2(i)*C1(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,1,5)= (B3(i)*B1(i)*CONST1/DIV_CONST1 + C3(i)*C1(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,1,6)= (C3(i)*B1(i)*CONST2/DIV_CONST1 + B3(i)*C1(i)*CONST3/DIV_CONST1)*hA

```



```

LOCAL_TS_MATRIX(i,2,1)= ((C1(i)*B1(i))/DIV_CONST1*(CONST2 + CONST3))*hA
LOCAL_TS_MATRIX(i,2,2)= (C1(i)**2*CONST1/DIV_CONST1 + B1(i)**2*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,2,3)= (B2(i)*C1(i)*CONST2/DIV_CONST1 + C2(i)*B1(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,2,4)= (C1(i)*C2(i)*CONST1/DIV_CONST1 + B1(i)*B2(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,2,5)= (C1(i)*B3(i)*CONST2/DIV_CONST1 + B1(i)*C3(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,2,6)= (C1(i)*C3(i)*CONST1/DIV_CONST1 + B1(i)*B3(i)*CONST3/DIV_CONST1)*hA

LOCAL_TS_MATRIX(i,3,1)= (B1(i)*B2(i)*CONST1/DIV_CONST1 + C2(i)*C1(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,3,2)= (B2(i)*C1(i)*CONST2/DIV_CONST1 + B1(i)*C2(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,3,3)= (B2(i)**2*CONST1/DIV_CONST1 + C2(i)**2*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,3,4)= (B2(i)*C2(i)*CONST2/DIV_CONST1 + B2(i)*C2(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,3,5)= (B2(i)*B3(i)*CONST1/DIV_CONST1 + C3(i)*C2(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,3,6)= (B2(i)*C3(i)*CONST2/DIV_CONST1 + B3(i)*C2(i)*CONST3/DIV_CONST1)*hA

LOCAL_TS_MATRIX(i,4,1)= (B1(i)*C2(i)*CONST2/DIV_CONST1 + B2(i)*C1(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,4,2)= (C1(i)*C2(i)*CONST1/DIV_CONST1 + B2(i)*B1(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,4,3)= (C2(i)*B2(i)/DIV_CONST1*(CONST2 + CONST3))*hA
LOCAL_TS_MATRIX(i,4,4)= (C2(i)**2*CONST1/DIV_CONST1 + B2(i)**2*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,4,5)= (B3(i)*C2(i)*CONST2/DIV_CONST1 + B2(i)*C3(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,4,6)= (C3(i)*C2(i)*CONST1/DIV_CONST1 + B2(i)*B3(i)*CONST3/DIV_CONST1)*hA

LOCAL_TS_MATRIX(i,5,1)= (B1(i)*B3(i)*CONST1/DIV_CONST1 + C1(i)*C3(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,5,2)= (C1(i)*B3(i)*CONST2/DIV_CONST1 + C3(i)*B1(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,5,3)= (B2(i)*B3(i)*CONST1/DIV_CONST1 + C3(i)*C2(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,5,4)= (C2(i)*B3(i)*CONST2/DIV_CONST1 + B2(i)*C3(i)*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,5,5)= (B3(i)**2*CONST1/DIV_CONST1 + C3(i)**2*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,5,6)= (B3(i)*C3(i)/DIV_CONST1*(CONST2 + CONST3))*hA

LOCAL_TS_MATRIX(i,6,1)= ((B1(i)*C3(i)*CONST2)/DIV_CONST1 + (B3(i)*C1(i)*CONST3)/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,6,2)= ((C1(i)*C3(i)*CONST1)/DIV_CONST1 + (B1(i)*B3(i)*CONST3)/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,6,3)= ((B2(i)*C3(i)*CONST2)/DIV_CONST1 + (C2(i)*B3(i)*CONST3)/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,6,4)= ((C2(i)*C3(i)*CONST1)/DIV_CONST1 + (B2(i)*B3(i)*CONST3)/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,6,5)= (((B3(i)*C3(i))*(CONST2 + CONST3))/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,6,6)= ((C3(i)**2*CONST1)/DIV_CONST1 + (B3(i)**2*CONST3)/DIV_CONST1)*hA

```

3.7. MATRIZ GLOBAL DE RIGIDEZ (MECÂNICO)

O algoritmo para gerar a matriz global de rigidez de forma automática a partir das matrizes de cada elemento é mostrado no **PARTE 7**.

Parte 7. Algoritmo da matriz de rigidez global (Análise Mecânica)

```

|*****
|
|      !CALCULA MATRIZ GLOBAL
|*****
ALLOCATE (G_TS_MATRIX(2*NODE_NUMBER, 2*NODE_NUMBER)); G_TS_MATRIX=0.D0
ALLOCATE (G_TS_MATRIX_BACKUP(2*NODE_NUMBER, 2*NODE_NUMBER)); G_TS_MATRIX=0.D0

DO I=1, ELEM_NUMBER
DO J=1, 3 !NUMERO DE NOS DO ELEMENTO
DO K=1, 2 ! NUMERO DE GRAUS DE LIBERDADE DO NO

      DO M=1,3 !NUMERO DE NOS DO ELEMENTO
      DO N= 1,2 !NUMERO DE GRAUS DE LIBERDADE

          LIN_G=2*(ELEM_NODE(i,j)-1)+K
          COL_G=2*(ELEM_NODE(i,M)-1)+N

          LIN_L=2*(j-1)+k
          COL_L= 2*(M-1)+N

          G_TS_MATRIX(LIN_G,COL_G)=G_TS_MATRIX(LIN_G,COL_G)+LOCAL_TS_MATRIX(i,LIN_L,COL_L )
          G_TS_MATRIX_BACKUP(LIN_G,COL_G)=G_TS_MATRIX(LIN_G,COL_G)

      END DO
      END DO
END DO
END DO
END DO

```

3.8. CONDIÇÕES DE CONTORNO

Após a criação da matriz global de rigidez do problema mecânico, deve-se aplicar as condições de contorno. No caso do problema mecânico, as condições de contorno podem ser de restrição ao movimento ou de imposição de deslocamento nodal. Em relação às forças, neste trabalho está sendo considerado unicamente as forças de origem térmica como condição de forças iniciais. Dessa forma, a rotina para implementar as condições de contorno é apresentada na **PARTE 8**.

Parte 8. Implementação das condições mecânicas de contorno

```

|*****
!aplicacao das restricoes na matriz global
|*****

DO i=1, NODE_NUMBER
do j=1, 2
  IF(NODE_RESTR(i,j)==1) THEN
    if(j==1)G_TS_MATRIX(2*(i-1)+1,:)=0.D0
    if(j==1)G_TS_MATRIX(:,2*(i-1)+1)=0.D0
    if(j==1)G_TS_MATRIX(2*(i-1)+1,2*(i-1)+1)=1.0D0

    if(j==2)G_TS_MATRIX(2*(i-1)+2,:)=0.D0
    if(j==2)G_TS_MATRIX(:,2*(i-1)+2)=0.D0
    if(j==2)G_TS_MATRIX(2*(i-1)+2,2*(i-1)+2)=1.0D0
  END IF
END DO
end do

!*****
!VETOR GLOBAL DE FORÇAS
!*****
ALLOCATE(G_FORCE_VECTOR(2*NODE_NUMBER));G_FORCE_VECTOR=0.D0
ALLOCATE(G_DISP_VECTOR(2*NODE_NUMBER));G_DISP_VECTOR=0.D0

DO I=1,ELEM_NUMBER
  n=1

  do j=1,3
    do k=1,2

      G_FORCE_VECTOR(2*(ELEM_NODE(i,j)-1)+k)= L_re_VETOR(i,n) + G_FORCE_VECTOR(2*(ELEM_NODE(i,j)-1)+k)

      n=n+1

    END DO
  end do
end do

!*****
DO i=1, NODE_NUMBER
do j=1, 2
  IF(NODE_RESTR(i,j)==1) THEN

    if(j==1)G_DISP_VECTOR(2*(i-1)+1)=0.D0 !APLICANDO RESTRIÇÃO
    if(j==2)G_DISP_VECTOR(2*(i-1)+2)=0.D0 !APLICANDO RESTRIÇÃO
  END IF
end do
end do

```

3.9. RESULTADO – DESLOCAMENTOS NODAIS

Como as sub-rotinas utilizadas para inverter matrizes (ludcmp e lubksb) alocam o resultado no próprio vetor de força inserido, aloca-se o vetor de forças nodais no vetor de deslocamento, para que no final, o deslocamento seja alocado no vetor correto, conforme é apresentado na **PARTE 9**.

Parte 9. Rotina para obtenção dos deslocamentos nodais

`G_DISP_VECTOR=G_FORCE_VECTOR`

```

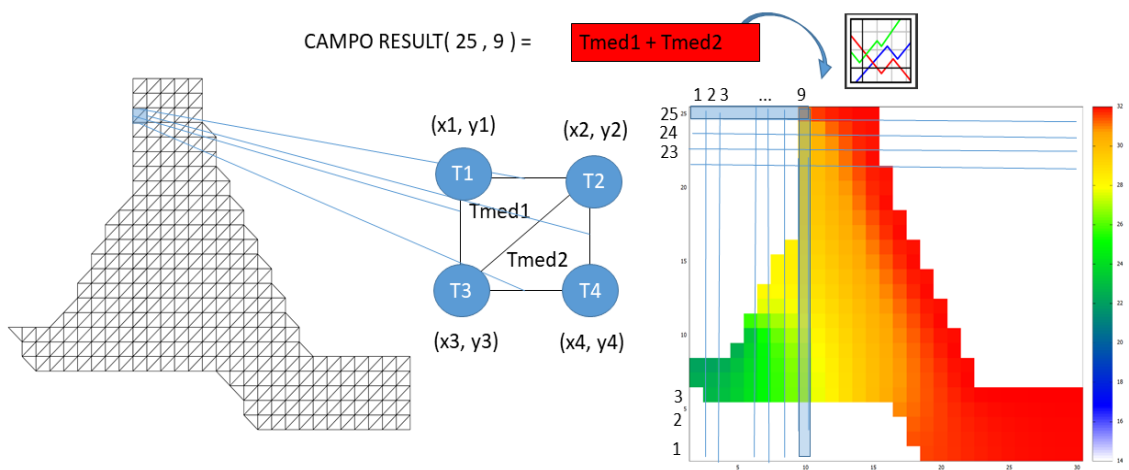
|*****
|
|SUBROTINAS QUE RESOLVEM O SISTEMA LU
|*****
allocate (THERMAL_STRESS_INDX(2*NODE_NUMBER))
call ludcmp(G_TS_MATRIX,2*NODE_NUMBER,2*NODE_NUMBER,THERMAL_STRESS_INDX,ddd)
call lubksb(G_TS_MATRIX,2*NODE_NUMBER,2*NODE_NUMBER,THERMAL_STRESS_INDX,G_DISP_VECTOR)

```

3.10. PÓS-PROCESSAMENTO

Afim de comparar os resultados do código desenvolvido e os resultados das análises realizadas utilizando o programa ANSYS®, desenvolveu-se uma rotina que calcula os valores de temperatura e deslocamento médio de cada elemento. Posteriormente, os resultados são alocados em uma matriz de acordo com a localização cartesiana do elemento no domínio bidimensional. O código gera um arquivo .txt de saída com a matriz de resultados. Finalmente, utiliza-se o programa *GNU PLOT* unicamente para converter os valores da matriz em cores, conforme ilustra a **Figura 19**.

Figura 19. Ilustração do pós-processamento



Para criar a matriz de resultados, determina-se o número de linhas e colunas da matriz de acordo com a quantidade de elementos, na horizontal e na vertical. Posteriormente, aloca-se o valor de temperatura ou deslocamento do elemento de acordo com a localização cartesiana dos nós que compõe os elementos, conforme é apresentado na **PARTE 10** do código.

Parte 10. Rotina para gerar a matriz de resultado

```

|*****
! ROTINA PARA GERAR GRÁFICOS COLORIDOS NO GNOPLT
|*****
X_MAX=0.D0;Y_MAX=0.D0

do i=1, NODE_NUMBER
  if(X_MAX < NODE_POSITION(i,1))X_MAX=NODE_POSITION(i,1)
  if(Y_MAX < NODE_POSITION(i,2))Y_MAX=NODE_POSITION(i,2)
end do

  X_MAX= X_MAX/STAND_X
  Y_MAX= Y_MAX/STAND_Y
|*****
!GERANDO GRÁFICOS - CAMPO DE TEMPERATURA
|*****
!allocate(MATRIX_VISUAL_2D(X_MAX,Y_MAX)); MATRIX_VISUAL_2D=100.0d0 !!!!! !EXEMPLO DE VALIDAÇÃO
allocate(MATRIX_VISUAL_2D(X_MAX,Y_MAX)); MATRIX_VISUAL_2D=14.0d0 !!!!! !BLOCO E6 INVERNO
!allocate(MATRIX_VISUAL_2D(X_MAX,Y_MAX)); MATRIX_VISUAL_2D=20.0d0 !!!!! !BLOCO E6 VERÃO

do i=1, ELEM_NUMBER

  X= NODE_POSITION(ELEM_NODE(i,3),1)/STAND_X
  Y= NODE_POSITION(ELEM_NODE(i,3),2)/STAND_Y

  AVE_THERMAL_ELEM=(VETOR_TEMP_NODE(ELEM_NODE(i,1))+VETOR_TEMP_NODE(ELEM_NODE(i,2))+VETO
R_TEMP_NODE(ELEM_NODE(i,3)))/3

  MATRIX_VISUAL_2D(X,Y)=AVE_THERMAL_ELEM

end do
|*****
!SAIDA_MATRIX_GRAPHIC.TXT - CADA POSIÇÃO MATRICIAL REPRESENTA UM ELEMENTO
|*****
open(unit=2,file="SAIDA1_THERMAL_GRAPHIC_MATRIX.txt")
do i=1, Y_MAX
  write(2,*)(MATRIX_VISUAL_2D(j,i),j=1,X_MAX)
  !write(*,*)(MATRIX_VISUAL_2D(i,j),j=1,X_MAX)
end do
close(2)
|*****
!GERANDO GRÁFICOS - CAMPO DE DESLOCAMENTOS EM X
|*****
MATRIX_VISUAL_2D=0.0005D0 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!INVERNO
!MATRIX_VISUAL_2D=-0.0005D0 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!VERÃO
!MATRIX_VISUAL_2D=0.d0 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!EXEMPLO DE VALIDAÇÃO

do i=1, ELEM_NUMBER

  X= NODE_POSITION(ELEM_NODE(i,3),1)/STAND_X
  Y= NODE_POSITION(ELEM_NODE(i,3),2)/STAND_Y

  AVE_DISP_ELEM=(G_DISP_VECTOR(2*(ELEM_NODE(i,1)-1)+1)+G_DISP_VECTOR(2*(ELEM_NODE(i,2)-
1)+1)+G_DISP_VECTOR(2*(ELEM_NODE(i,3)-1)+1))/3

  MATRIX_VISUAL_2D(X,Y)= AVE_DISP_ELEM

END DO
|*****
!SAIDA_MATRIX_GRAPHIC.TXT - CADA POSIÇÃO MATRICIAL REPRESENTA UM ELEMENTO
|*****
open(unit=3,file="SAIDA2_X_TS_DISP_GRAPHIC_MATRIX.txt")

```

```

do i=1, Y_MAX
  write(3,*)(MATRIX_VISUAL_2D(j,i),j=1,X_MAX)
end do
close(3)
!*****
!GERANDO GRÁFICOS - CAMPO DE DESLOCAMENTOS EM Y
!*****
MATRIX_VISUAL_2D=2.0D0 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!INVERNO
MATRIX_VISUAL_2D=-0.01D0 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!VERÃO
!MATRIX_VISUAL_2D=0.40 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!EXEMPLO DE VALIDAÇÃO

do i=1, ELEM_NUMBER

  X= NODE_POSITION(ELEM_NODE(i,3),1)/STAND_X
  Y= NODE_POSITION(ELEM_NODE(i,3),2)/STAND_Y

AVE_DISP_ELEM=(G_DISP_VECTOR(2*(ELEM_NODE(i,1)-1)+2)+G_DISP_VECTOR(2*(ELEM_NODE(i,2)-
1)+2)+G_DISP_VECTOR(2*(ELEM_NODE(i,3)-1)+2))/3

  MATRIX_VISUAL_2D(X,Y)= AVE_DISP_ELEM

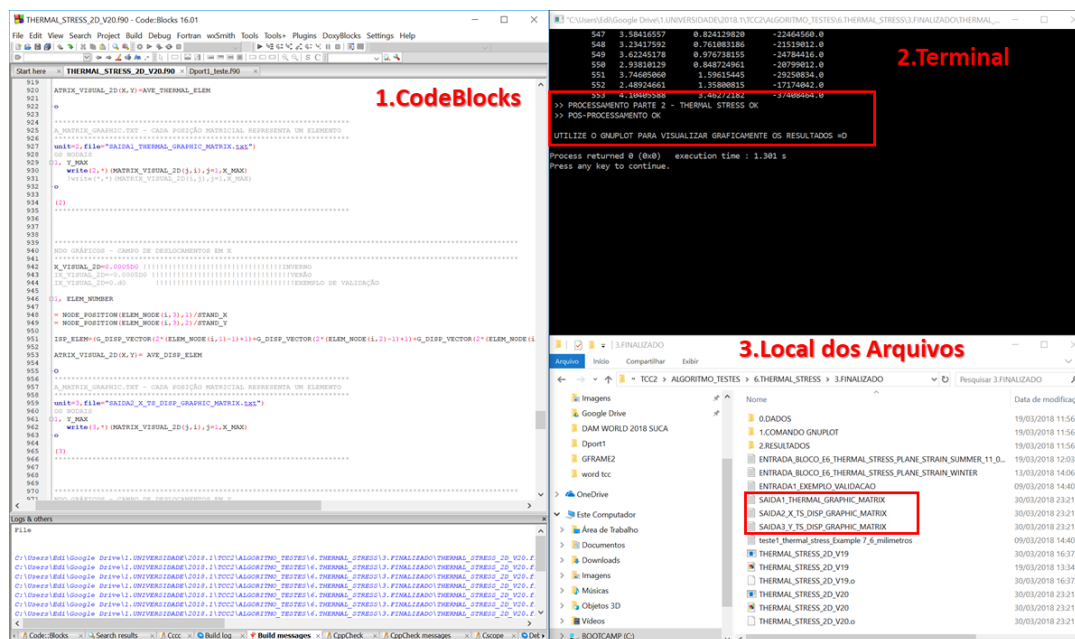
END DO
!*****
!SAIDA_MATRIX_GRAPHIC.TXT - CADA POSIÇÃO MATRICIAL REPRESENTA UM ELEMENTO
!*****
open(unit=4,file="SAIDA3_Y_TS_DISP_GRAPHIC_MATRIX.txt")
do i=1, Y_MAX
  write(4,*)(MATRIX_VISUAL_2D(j,i),j=1,X_MAX)
end do

close(4)
!*****

```

Os resultados são gravados em arquivos de texto de saída que são criados assim que o algoritmo é compilado, conforme a **Figura 20**.

Figura 20. Criação de arquivos de texto de saída para plotar resultados

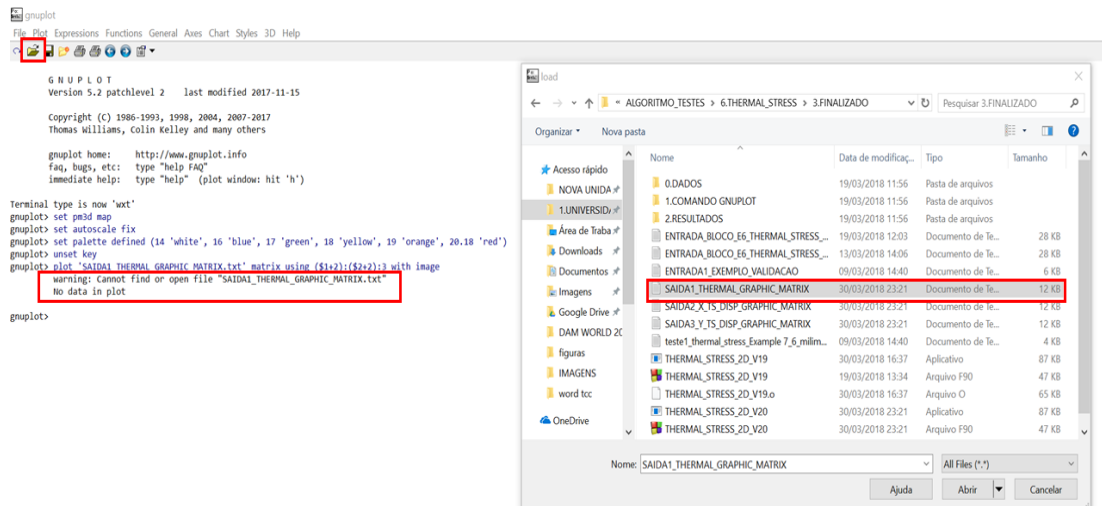


Para plotar a matriz de resultados no programa GNUPLOT, deve-se configurar a escala e as cores do gráfico. Em seguida, deve-se localizar o arquivo através da ferramenta 'abrir arquivo' da barra de ferramenta da interface inicial do programa

GNU PLOT. No caso da barragem de ITAIPU, considerando a condições de verão, os comandos de ajuste de plotagem são os apresentados na **Figura 21**.

Todos os comandos de ajustes do utilizados neste trabalho são apresentados no **APÊNDICE III**.

Figura 21. Comandos de ajustes no *GNU PLOT*



Utilizando os comandos apresentados abaixo, obtém-se o resultado gráfico do campo de temperatura, conforme a **Erro! Fonte de referência não encontrada..**

set pm3d map

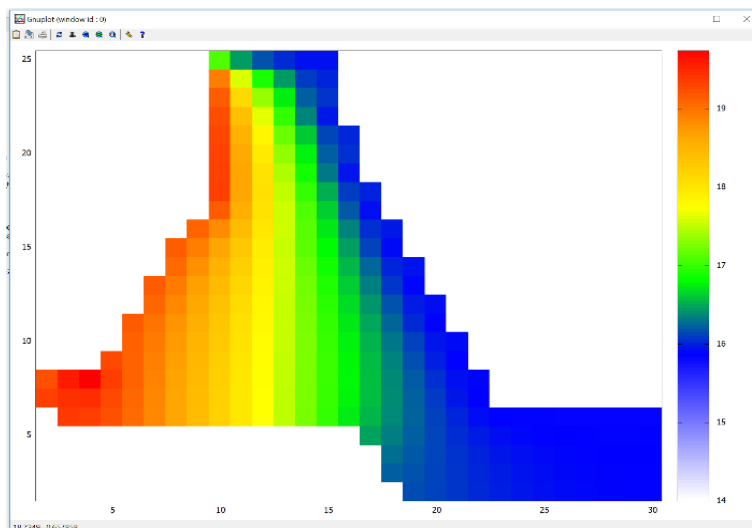
set autoscale fix

set palette defined (14 'white', 16 'blue', 17 'green', 18 'yellow', 19 'orange', 20.18 'red')

unset key

plot 'SAIDA1_THERMAL_GRAPHIC_MATRIX.txt' matrix using (\$1+2):(\$2+2):3 with image

Figura 22. Plotagem de resultados utilizando o programa *GNU PLOT*

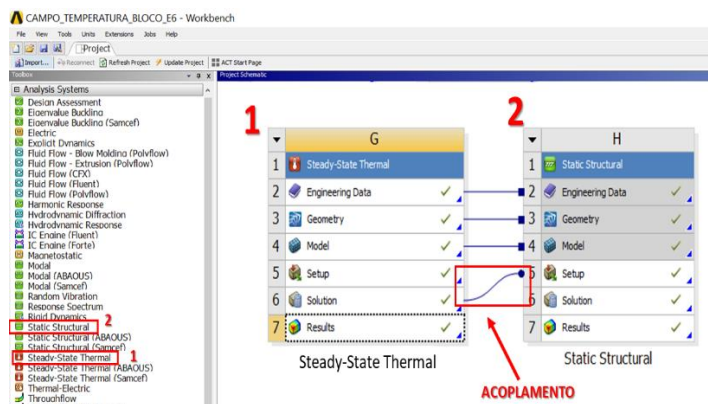


4. ACOPLAMENTO TÉRMO-MECÂNICO UTILIZANDO ANSYS®

O *WorkBench* é o ambiente desenvolvido pela ANSYS® para agrupar vários pacotes de análise e simulação computacional. Esse ambiente permite realizar acoplamentos, de forma fácil e intuitiva, entre as simulações permitindo usar o resultado de uma análise como condição inicial de outra análise.

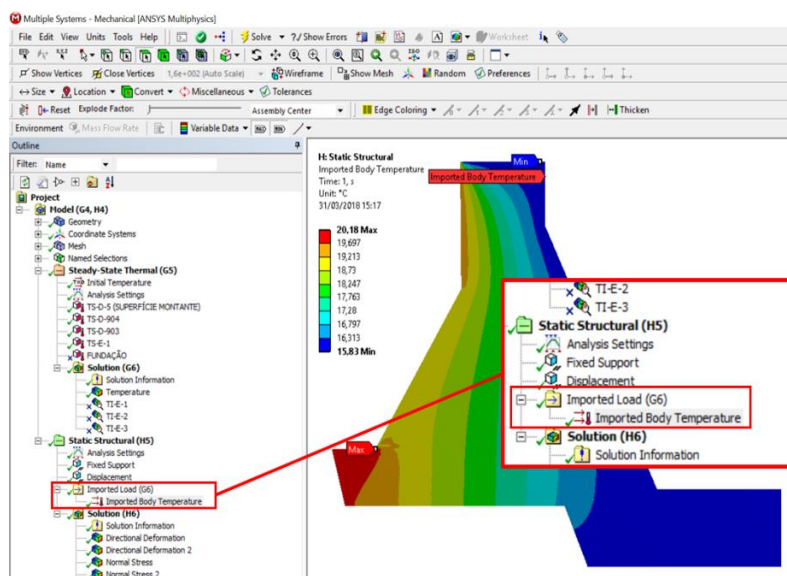
Após realizar uma simulação térmica estacionária, basta arrastar o ícone de uma análise estática estrutural e soltar em cima da caixa de resultado da simulação térmica, conforme a **Figura 23**.

Figura 23. Procedimento para acoplamento de análises no Workbench (Ansys®)



Automaticamente, o ambiente Workbench inicia uma análise estática estrutural utilizando o campo de temperatura resultante como condição inicial de carregamento, conforme a **Figura 24**.

Figura 24. Campo de temperatura como condição de inicial da análise estática

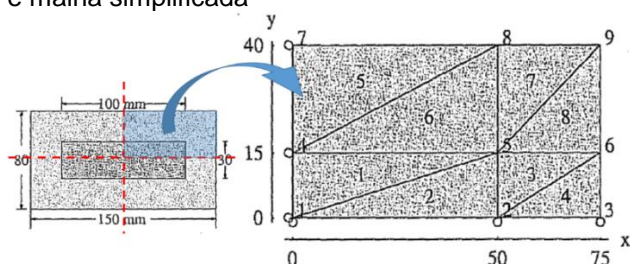


5. VALIDAÇÃO DO ALGORITMO

Inicialmente, calibrou-se o código aplicando vários exercícios resolvidos e comparando os resultados.

Um dos problemas de validação do algoritmo foi a análise térmica estrutural de uma placa de aço e alumínio, ilustrada na **Figura 25**. Devido a dupla simetria, é possível simplificar o problema.

Figura 25. Placa de alumínio e aço submetida à variação de temperatura e malha simplificada



Nesse caso, considerou-se uma variação de temperatura uniforme em todo o domínio de 70 °C. As propriedades dos materiais são apresentadas na **Tabela 2**.

Tabela 2. Propriedades dos materiais do problema de validação

Material	Aço	Alumínio
Módulo de Elasticidade (E)	200 [GPa]	70 [GPa]
Coefficiente de Poisson (ν)	0.3	0.33
Coefficiente de Dilatação térmica (α)	$12 \times 10^{-6} [^{\circ}\text{C}]$	$23 \times 10^{-6} [^{\circ}\text{C}]$

Inicialmente a chapa não possui restrição ao movimento. Porém, no modelo simplificado, com um quarto do domínio, deve-se considerar condições de contorno para representar o comportamento real. Na **Figura 26** são apresentadas as condições essenciais de contorno do problema. A letra u representa o descolamento horizontal e v o deslocamento vertical.

Figura 26. Condições de restrição ao movimento

Node	dof	Value
1	u_1	0
	v_1	0
2	v_2	0
3	v_3	0
4	u_4	0
7	u_7	0

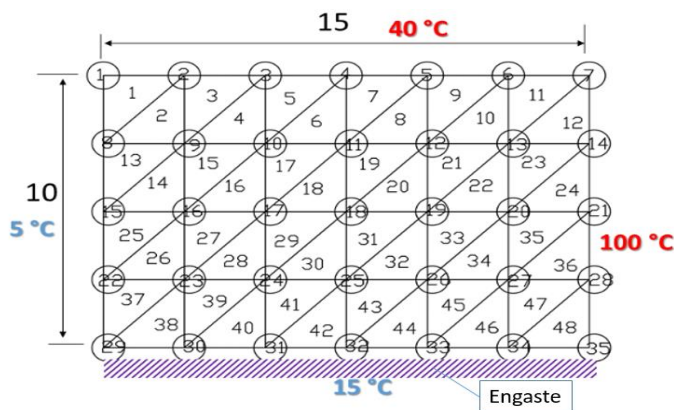
Os resultados disponíveis na literatura bem como os resultados do algoritmo são apresentados na **Figura 27**.

Figura 27. Comparação dos resultados de deslocamentos nodais do problema de validação

RESULTADO DO LIVRO	RESULTADO DO ALGORITMO
$u_2 = 0.0513447$	GL DIR DESL.
$u_3 = 0.0703132$	1 U1 0.00000000
$v_4 = 0.0252714$	2 V1 0.00000000
$u_5 = 0.0495551$	3 U2 5.13446666E-02
$v_5 = 0.0186102$	4 V2 0.00000000
$u_6 = 0.069253$	5 U3 7.03131407E-02
$v_6 = 0.0146016$	6 V3 0.00000000
$v_7 = 0.0445986$	7 U4 0.00000000
$u_8 = 0.0498168$	8 V4 2.52714176E-02
$v_8 = 0.0388815$	9 U5 4.95550483E-02
$u_9 = 0.0716126$	10 V5 1.86101999E-02
$v_9 = 0.0366734$	11 U6 6.92529604E-02
	12 V6 1.46016050E-02
	13 U7 0.00000000
	14 V7 4.45986502E-02
	15 U8 4.98167910E-02
	16 V8 3.88815440E-02
	17 U9 7.16125369E-02
	18 V9 3.66734453E-02

Após a validação do algoritmo, realizou-se a análise termo-mecânica de uma placa submetida a diferentes temperaturas e engastada em um dos vértices, conforme é ilustrado na **Figura 28**. A mesma análise utilizando o programa Ansys®17.2 foi realizada para verificação da convergência do resultado.

Figura 28. Placa engastada submetida a diferentes temperaturas nas bordas



Os resultados são apresentados nas **Figura 29**, **Figura 30**, **Figura 31**, **Figura 32**, **Figura 33** e **Figura 34**.

Figura 29. Campo de Temperatura (Algoritmo)

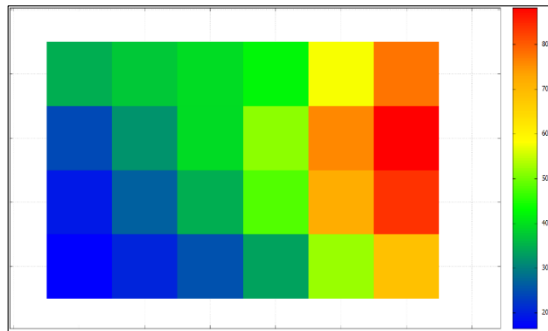


Figura 30. Campo de Temperatura (Ansys®)

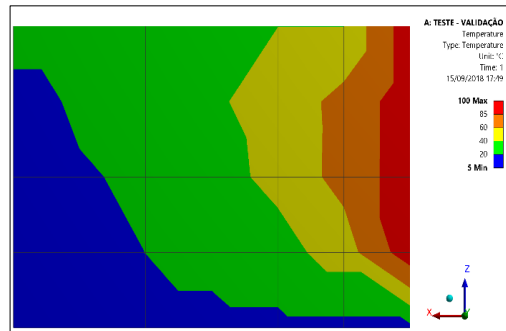


Figura 31. Campo de Deslocamento Vertical. (Algoritmo)

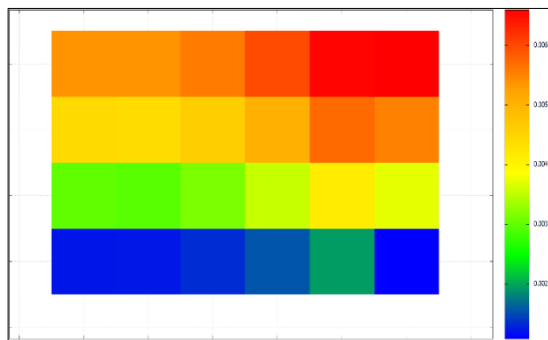


Figura 32. Campo de Deslocamento Vertical. (Ansys®)

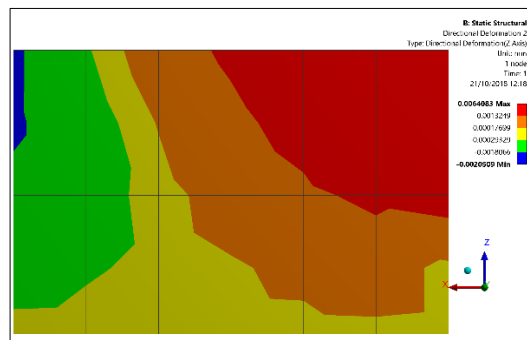


Figura 33. Campo de Deslocamento Horizontal. (Algoritmo)

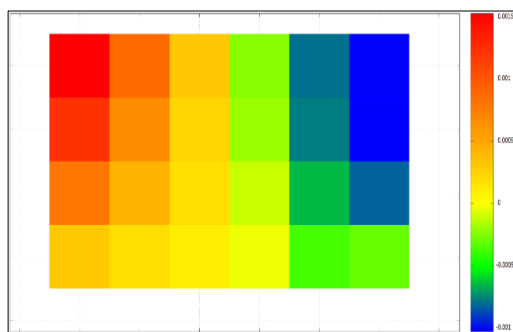
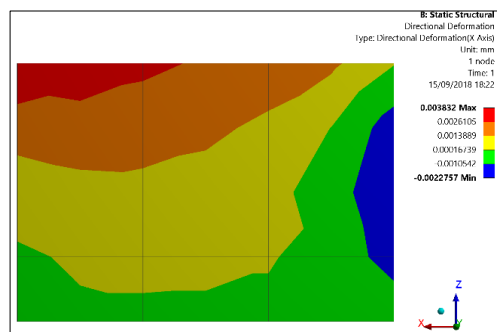


Figura 34. Campo de Deslocamento Horizontal. (Ansys®)



6. ESTUDO DE CASO - CENTRAL HIDRELÉTRICA DE ITAIPU (CHI)

Para se avaliar o potencial do algoritmo de análise térmica-estrutural desenvolvido neste trabalho, realizou-se uma simulação térmica em um bloco de contraforte da CHI. Os resultados de temperaturas internas, bem como o deslocamento da crista da barragem foram comparados com dados de monitoramento dos instrumentos desta. Neste capítulo é realizada uma breve introdução sobre a CHI, informações de projeto e os dados instrumentais que serão utilizados para a realização desta análise.

A Central Hidrelétrica de ITAIPU, é uma usina hidrelétrica binacional localizada no Rio Paraná, na fronteira entre o Brasil e o Paraguai. A barragem foi construída pelos dois países entre 1975 e 1982. Possui quase 8 Km de extensão e é formada por 9 barragens de vários tipos, sendo: de terra, enrocamento, contrafortes, gravidade e gravidade aliviada, conforme é mostrado na **Figura 35**.

Figura 35. Projeto arquitetônico geral da CHI

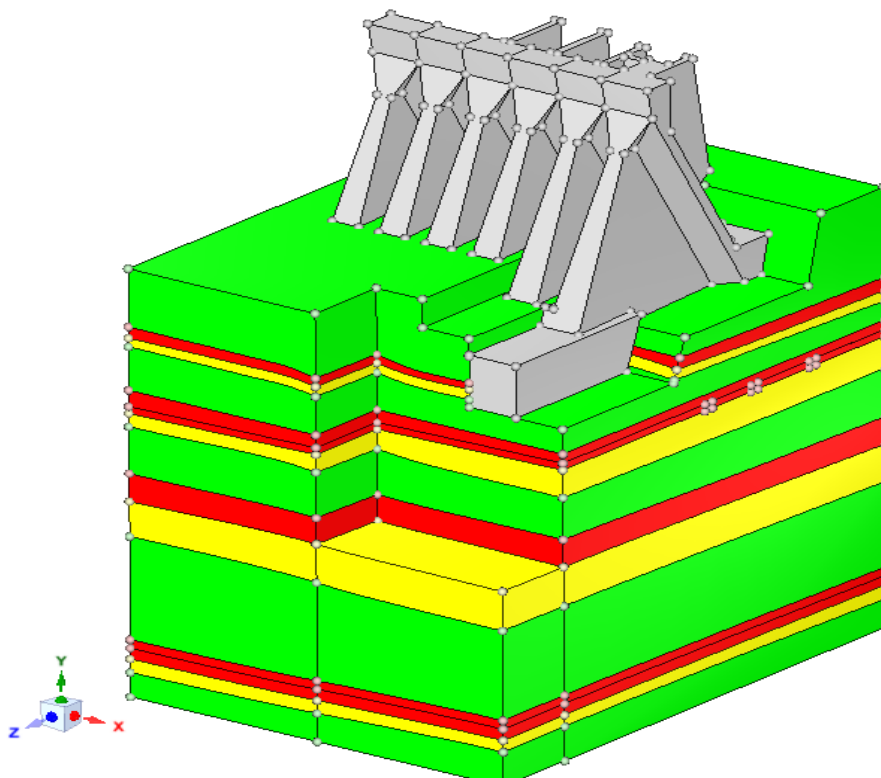


Fonte: ITAIPU (2017).

A barragem é dividida em: estrutura principal, estrutura de desvio, barragem de terra da margem direita, barragem de enrocamento e barragem de terra da margem esquerda. A barragem de contraforte de ligação direita é denominada Trecho E, localizado na ombreira direita, entre a Barragem Principal (Trecho F) e a Barragem Lateral Direita (Trecho D). É constituído por 6 blocos de contrafortes (do E1 ao E6),

apoiados em maciço basáltico. Na **Figura 36** é apresentado a vista isométrica do modelo tridimensional do Trecho E da CHI.

Figura 36. Trecho E da CHI



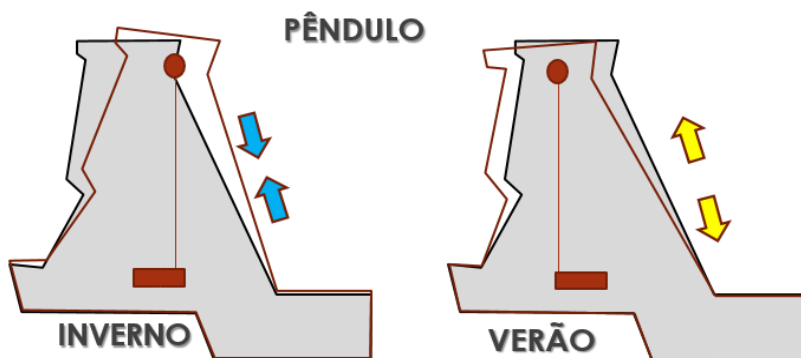
Fonte: CEASB-PTI

Para acompanhar o comportamento estrutural da barragem, esta possui mais de 2.792 instrumentos de monitoramento.

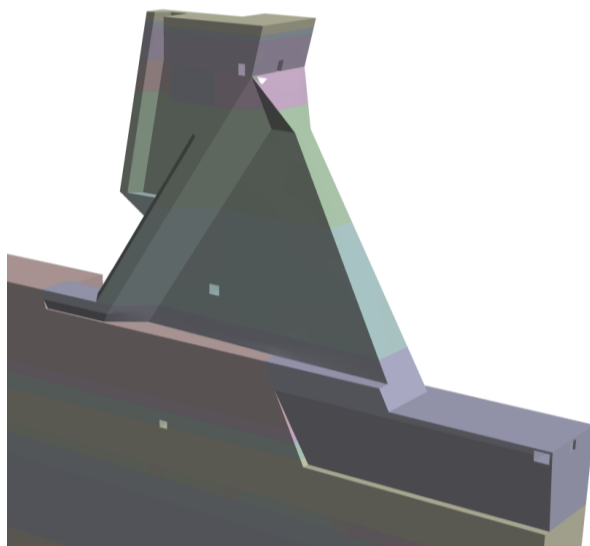
O bloco E-6 é o mais instrumentado e por isso, considerado um bloco chave de monitoramento. Possui quase todos os instrumentos, entre eles: o pêndulo direto, termômetros, piezômetros, tensômetros, entre outros.

Os termômetros podem ser internos (TI) ou de superfície (TS). Neste trabalho, utilizou-se os termômetros internos para se determinar as condições de temperatura de contorno na análise térmica. Já os termômetros internos foram utilizados para validação dos resultados.

O pêndulo tem como finalidade definir os deslocamentos em dois eixos ortogonais, radial e tangencial. Através desses deslocamentos é possível quantificar os esforços que a estrutura está sendo submetida, ao mesmo tempo que a relação entre duas leituras resultará no movimento absoluto de qualquer deslocamento de um ponto observado. Na **Figura 37** é ilustrado o movimento do pêndulo de acordo com o comportamento da barragem em diferentes estações do ano.

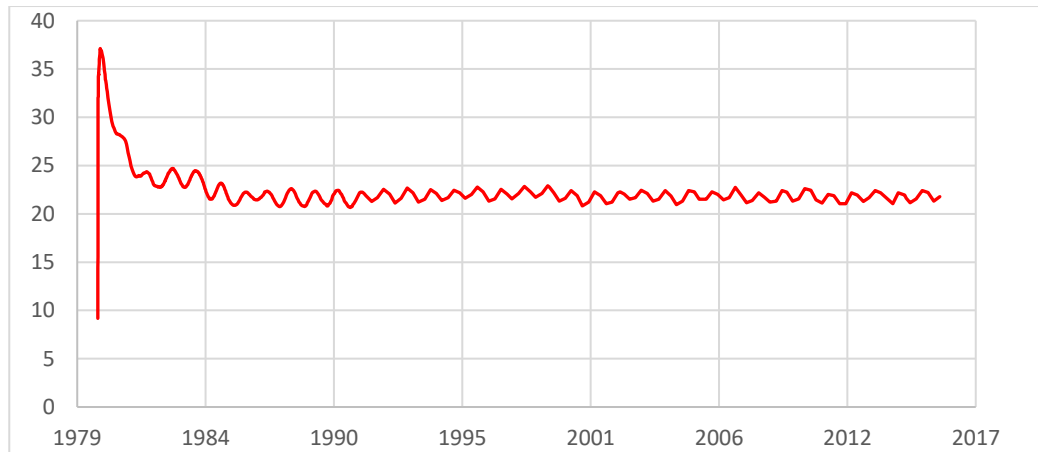
Figura 37. Ilustração do funcionamento do pêndulo

Para avaliar o comportamento desse trecho, e principalmente do Bloco E-6, diversos estudos foram realizados através de simulações matemáticas baseado no Método dos Elementos Finitos, com o intuito de reproduzir o desempenho do bloco ao ser submetido a diversos carregamentos, como por exemplo a análise de estabilidade sísmica desenvolvida por SILVA JUNIOR (2017). Na **Figura 38** é mostrado o perfil da seção transversal do Bloco E-6 e a localização dos principais instrumentos.

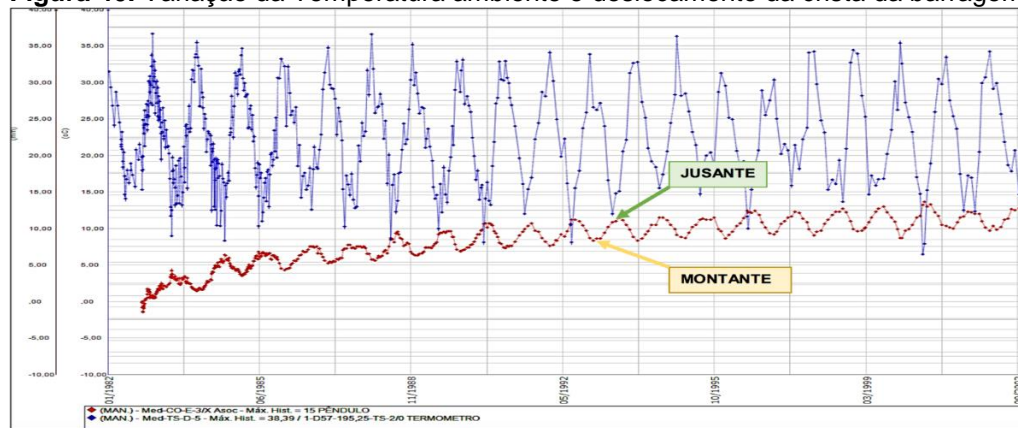
Figura 38. Bloco E-6 da CHI

Fonte: CEASB-PTI

Analisando os dados de um termômetro interno do bloco E-6, na **Figura 39** é possível visualizar o comportamento térmico da estrutura desde a fase de construção até. A elevada temperatura de 1979 a 1984 pode ser explicada pelo calor de hidratação do concreto nos primeiros anos de vida da estrutura. Posteriormente, a temperatura acompanha a variação sazonal do ambiente.

Figura 39. Variação da temperatura interna do bloco de contraforte E-6

Em relação ao comportamento estrutural da barragem, na **Figura 40** é ilustrado a evolução da temperatura sazonal do ambiente e o deslocamento da crista do Bloco. O tempo (de 1982 a 2002) é apresentado no eixo da abcissa. No eixo vertical há duas escalas, a primeira, refere-se à ao deslocamento horizontal em milímetros, a segunda, a temperatura em graus célsius. A curva azul representa a temperatura medida pelo termômetro, apresentando um comportamento senoidal com amplitudes térmicas que vão de 6 a 38 °C. A curva vermelha representa a medição do pendulo, que expressa o deslocamento horizontal da crista (topo da estrutura) entre a jusante e a montante.

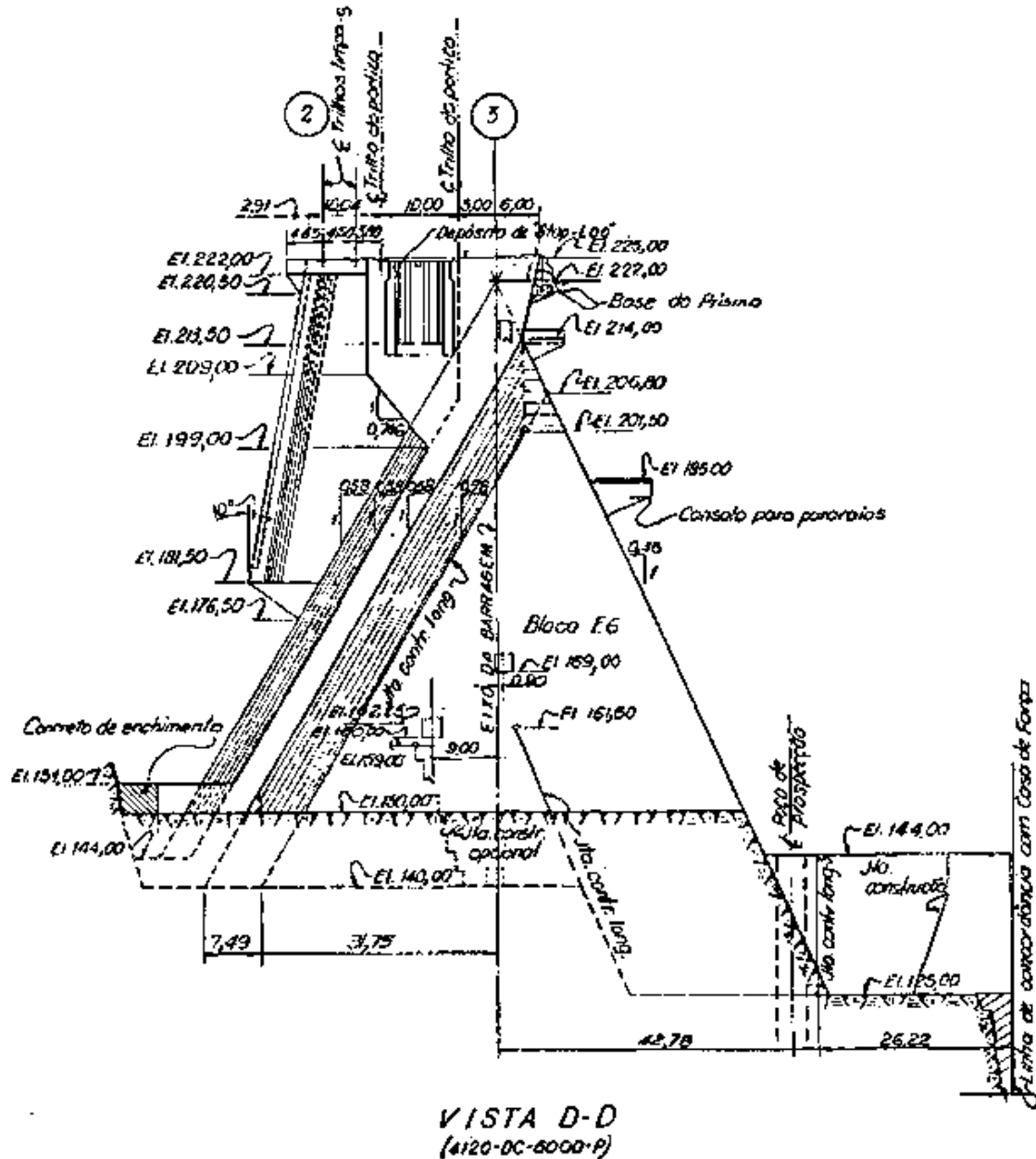
Figura 40. Variação da Temperatura ambiente e deslocamento da crista da barragem

Os instrumentos mostram que durante o inverno, a crista da barragem se desloca para jusante e no verão para montante. Esse comportamento é esperado, uma vez que durante o inverno, a temperatura ambiente diminui e a superfície da barragem a jusante sofre contração térmica. No verão, devido à elevação da temperatura e a incidência solar, o paramento de jusante sofre dilatação térmica, deslocando a barragem na direção de montante (ITAIPU, 2017).

6.1. DADOS GEOMÉTRICOS

Para o estudo de caso, a modelagem geométrica foi iniciada através dos desenhos de projetos utilizados na época de construção da barragem, conforme a **Figura 41**.

Figura 41. Vista Lateral do Bloco E-6

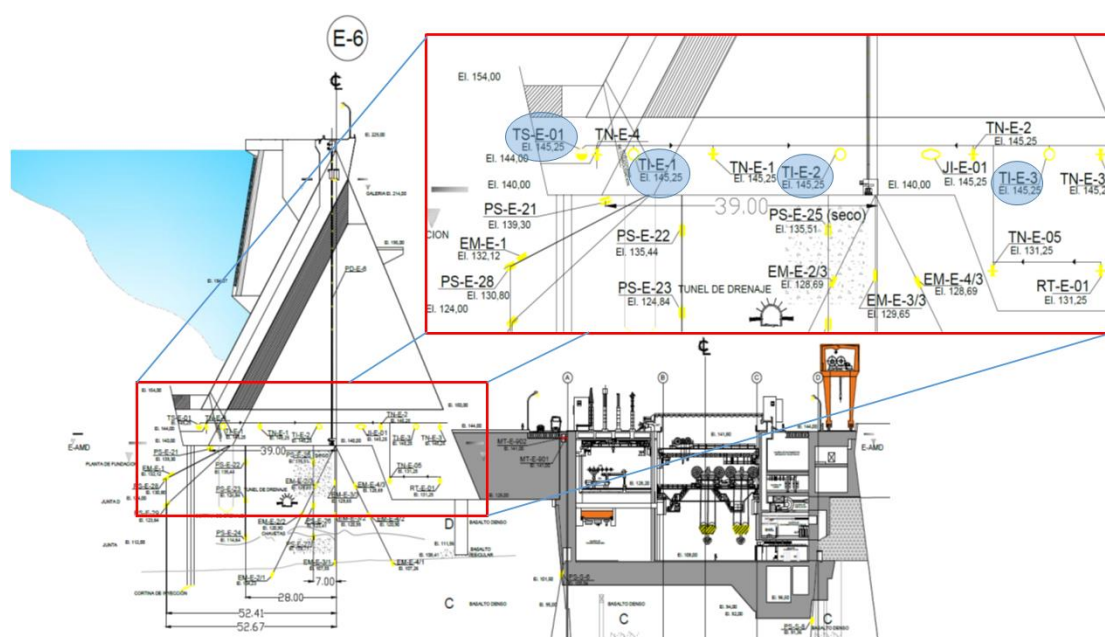


Fonte: ITAIPU (2017).

6.1.1. DADOS INSTRUMENTAIS

Na CHI os instrumentos são padronizados de acordo o tipo e a localização. Os termômetros superficiais (TS), estão instalados a aproximadamente quatro metros de profundidade em relação à superfície da barragem. Os termômetros internos (TI) foram instalados no interior da barragem, na região central próximo à fundação. Na **Figura 42** é possível observar a localização dos termômetros superficiais e internos localizados no Bloco E-6.

Figura 42. Localização dos termômetros do bloco E6.

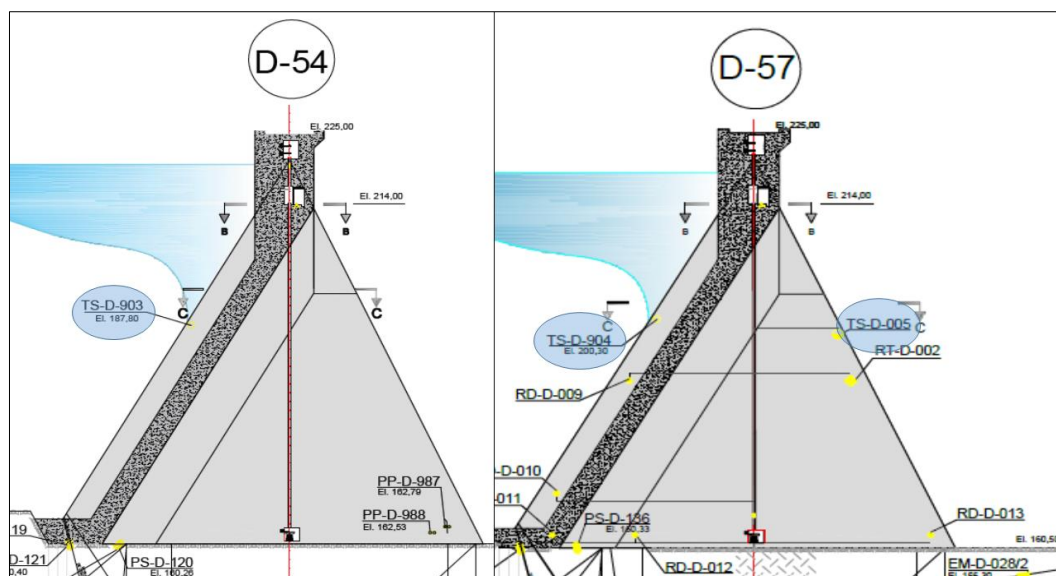


Fonte: Adaptado ITAIPU (2017).

As medições da temperatura no corpo da barragem são realizadas por termômetros distribuídos ao longo dos blocos periodicamente desde o período de construção até hoje. O Bloco E-6 possui 4 termômetros (TS-E-001, TI-E-001, TI-E-002 e TI-E-003), localizados na superfície e no interior da estrutura, próximos da fundação.

Infelizmente, o bloco E-6 não possui termômetros nas cotas mais elevadas da barragem. Para resolver esse problema, admitiu a utilização de dados de termômetros dos Blocos D-54 e D-57, devido à similaridade e a proximidade com bloco de interesse. Desse modo, o termômetro TS-D-903 posicionado a montante do Bloco D-54 e os termômetros TS-D-904 e TS-D-5, do Bloco D-57 são utilizados como condição de contorno da superfície de montante e de jusante do modelo matemático. A **Figura 43** mostra a localização dos termômetros utilizados desses blocos.

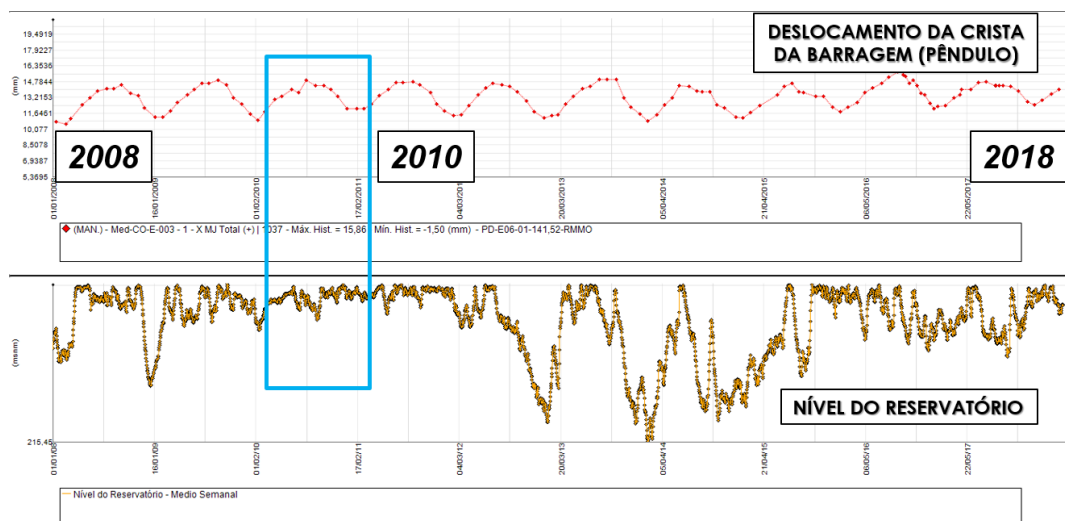
Figura 43. Localização dos termômetros dos Bloco D-54 e do D-57



Fonte: Adaptado ITAIPU (2017).

Neste trabalho, considerou-se unicamente os esforços internos provenientes da variação de temperatura como carregamento nas análises. Portanto, buscou-se o ano onde ocorreu a menor variação do nível do reservatório e, conseqüentemente, a menor variação da pressão hidrostática. Dessa forma, o comportamento mecânico da barragem ocorreu basicamente devido a variação de temperatura. Na **Figura 44** é apresentado o gráfico de variação do nível do reservatório e deslocamento do pêndulo do bloco E-6. Através desse gráfico concluiu-se que o ano de 2010 o reservatório apresentou a menor variação comparado com os demais anos.

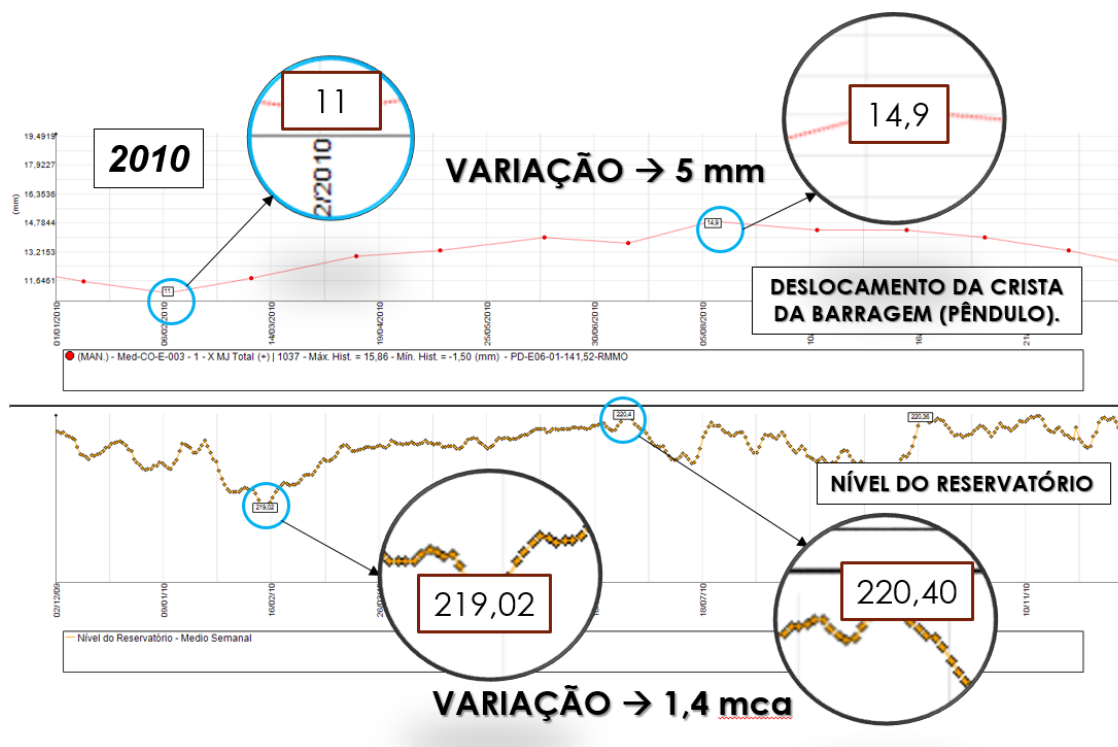
Figura 44. Variação do nível do reservatório e deslocamento da crista da barragem



Fonte: Adaptado ITAIPU (2017).

Os dados do pêndulo de 2010 são apresentados na **Figura 45**. Registrou-se uma variação total de aproximadamente 5 mm. Esse valor é muito importante para se realizar a validação dos resultados das simulações.

Figura 45. Variação do nível do reservatório e deslocamento da crista da barragem



Fonte: Adaptado ITAIPU (2017).

Finalmente, filtrou-se os dados os termômetros dos blocos D-54, D57 e E-6, apresentados na **Tabela 3**, para se determinar as temperaturas de contorno do modelo, tanto no verão, como no inverno.

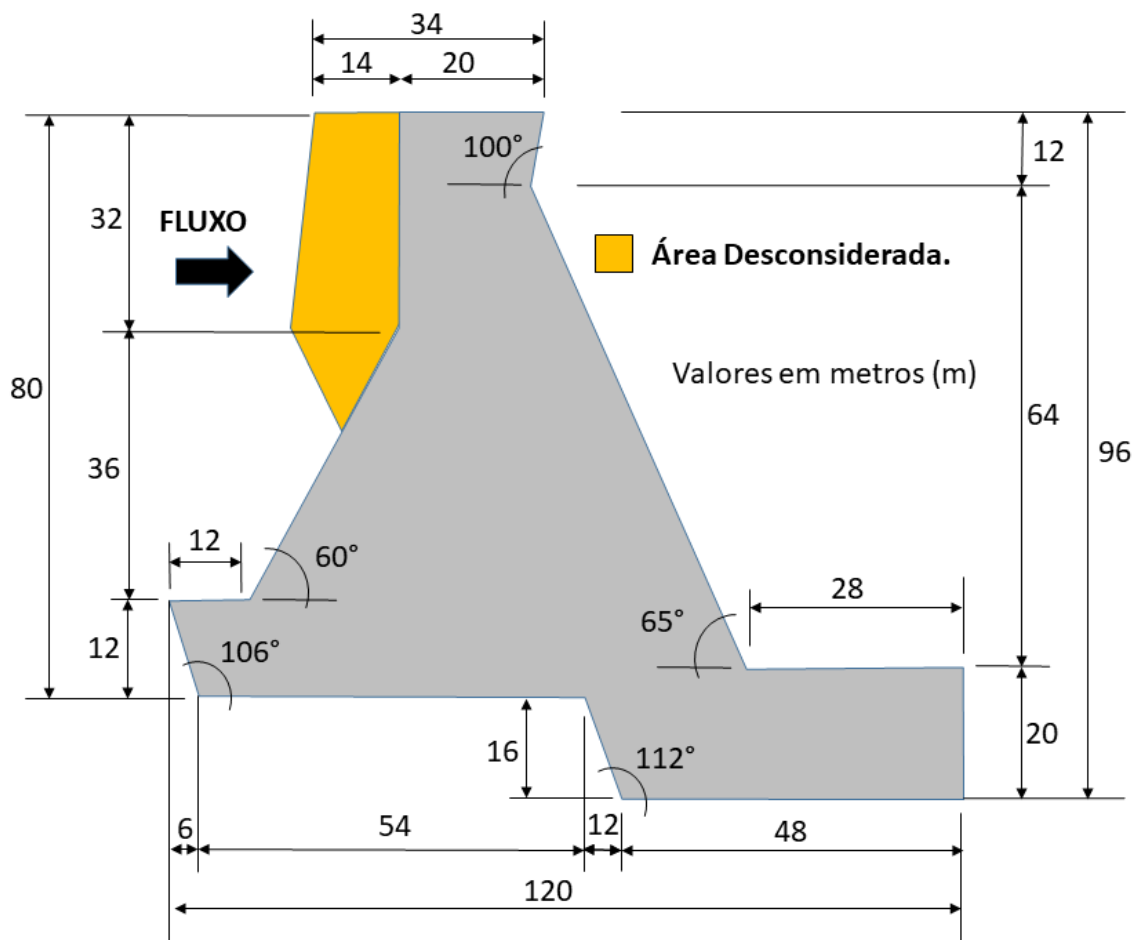
Tabela 3 – Temperaturas utilizadas nas condições de contorno

Termômetro	Inverno (°C)	Verão (°C)
	Data:12/07/2010	Data:11/01/2010
TS-D-005	18,46	32,33
TS-D-904	21,1	29,27
TS-D-903	20,76	28,62
TS-E-001	21,47	22,65

6.2. MODELO MATEMÁTICO

Com base em vários projetos arquitetônicos, desenvolveu-se um modelo bidimensional simplificado do Bloco E-6, ilustrado na **Figura 46**.

Figura 46. Geometria Bidimensional do bloco E6.

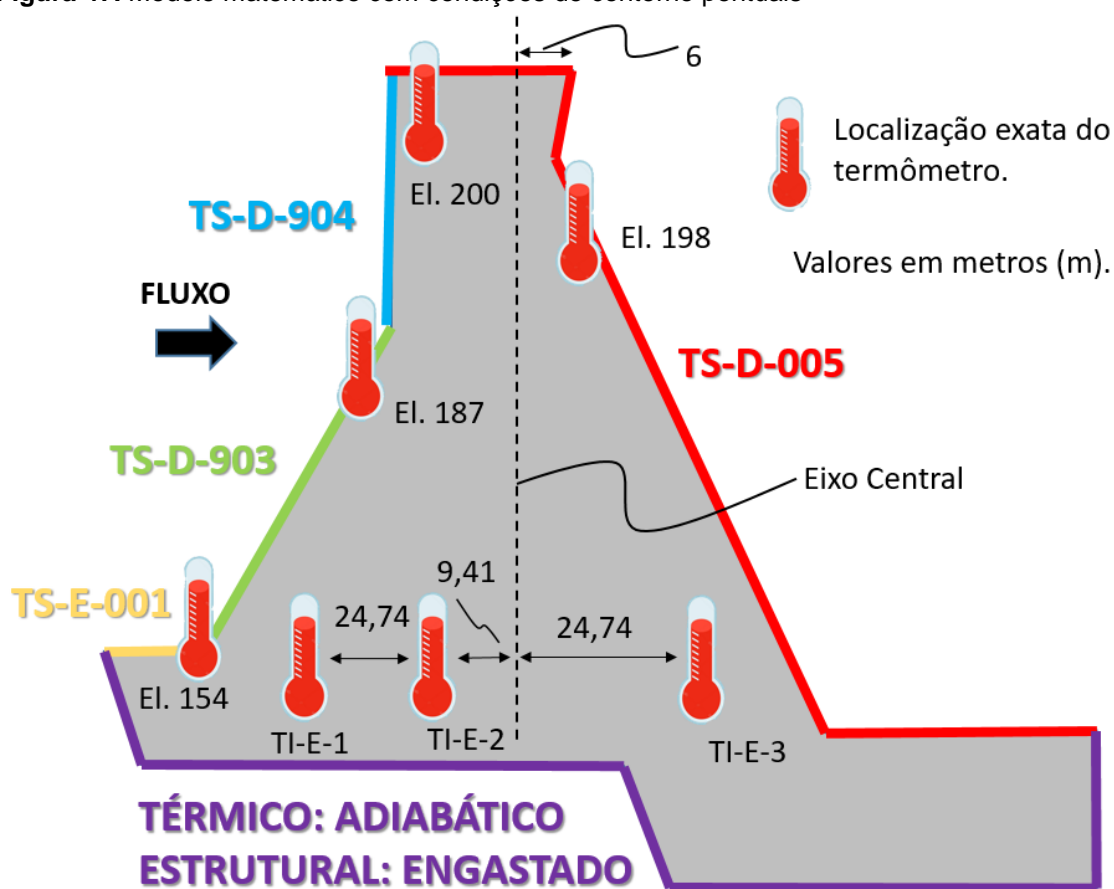


Como hipótese da análise mecânica bidimensional, adotou-se o estado plano de deformação, ou seja, considera-se que a superfície unitária analisada se encontra no meio de um domínio cujo comprimento perpendicular ao plano é extenso o suficiente para que as deformações perpendiculares ao plano sejam zero e as tensões sejam diferentes de zero. Dessa forma, desconsiderou-se a região em amarelo do modelo devido a sua esbeltez, espessura de aproximadamente 2,5 metros.

6.3. CONDIÇÕES DE CONTORNO

As condições de contorno foram definidas com os termômetros superficiais. Para fins de simplificação, assumiu-se a temperatura do termômetro de cada região para toda a superfície adjacente. A **Figura 47** ilustra as condições de contorno extrapoladas assumidas. Na aresta vermelha, considerou-se a temperatura do termômetro TS-D-005; na aresta azul, considerou-se a temperatura do TS-D-904; na aresta verde, considerou-se a temperatura da TS-D-903; na aresta laranja, considerou-se a temperatura do TS-E-001. Na aresta cor roxa, região de contato entre a barragem e a fundação, assumiu-se um comportamento adiabático devido à falta de informações das propriedades térmicas da fundação e porque não há instrumentos de monitoramento térmico nessa região. Em relação ao comportamento mecânico em função da variação de temperatura, a aresta roxa será engastada, ou seja, com restrição de movimento na horizontal e na vertical.

Figura 47. Modelo matemático com condições de contorno pontuais



6.4. MALHA

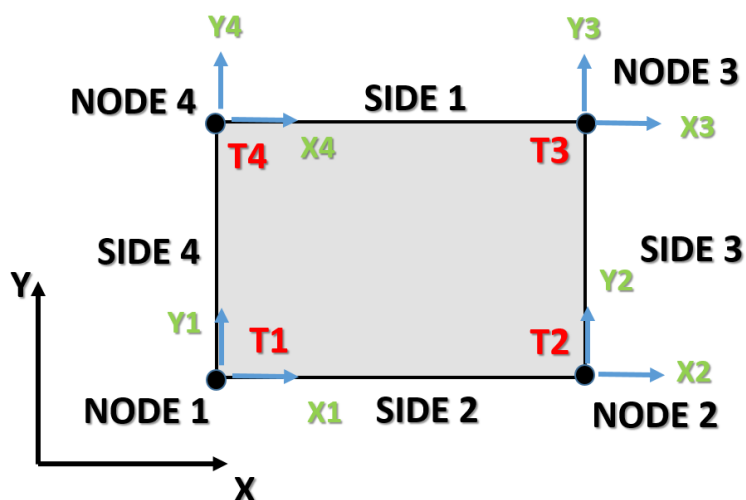
A malha foi criada com base no modelo apresentado na **Figura 46**. Para determinar as coordenadas nodais, utilizou-se o programa *AUTOCAD*.

Realizou-se um estudo de convergência de malha considerando dois tipos de elementos, retangular e triangular.

Além do vetor de resultados nodais, desenvolveu-se uma rotina que gera uma matriz de resultado, onde cada elemento matricial corresponde a uma parte do domínio bidimensional. Essa rotina, permite gerar campos de resultados que podem ser comparados com os resultados do programa Ansys®. A condição para que os resultados sejam alocados corretamente nessa matriz de resultado final é a uniformidade dos elementos. Por isso, ambas as malhas, formadas por elementos retangulares e triangulares, são uniformes.

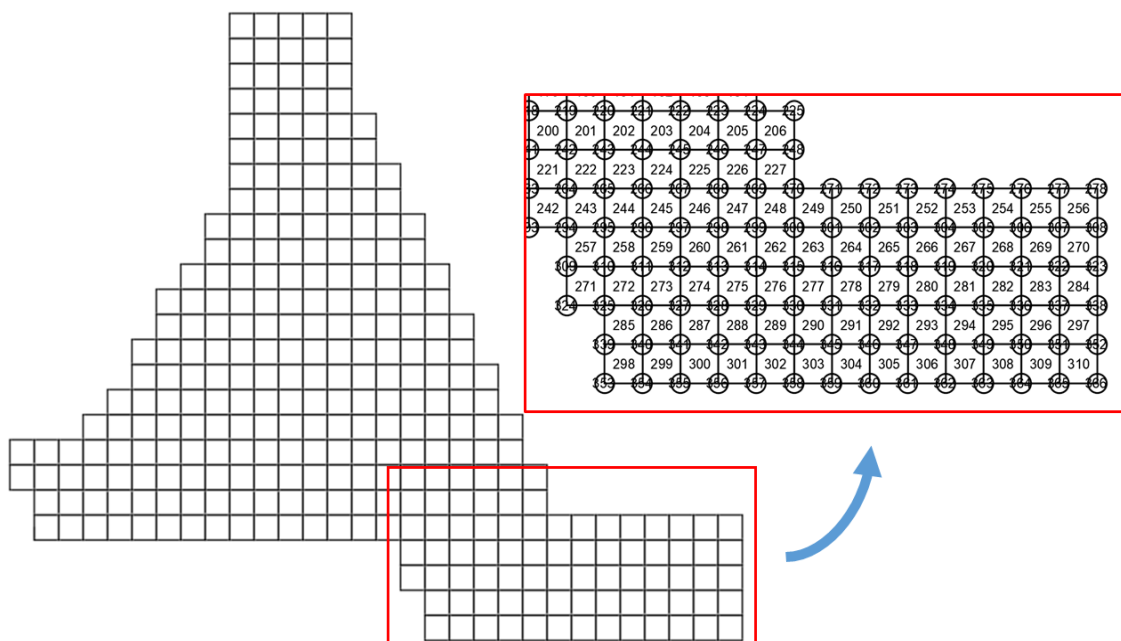
Na **Figura 49** é apresentado o elemento utilizado na geração da malha composta por elementos retangulares.

Figura 48. Ilustração do elemento retangular de quatro nós



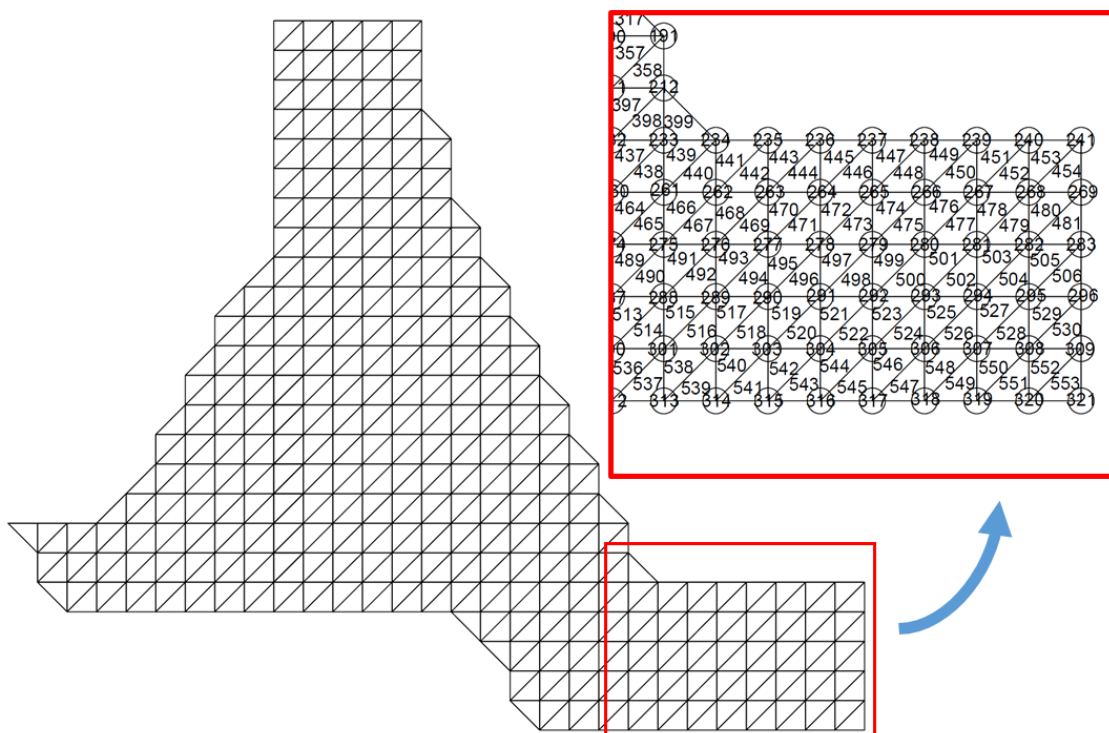
A malha formada por elementos retangulares, ilustrado na **Figura 49**, não apresentou bons nas arestas inclinadas de contorno, a montante e jusante do modelo.

Figura 49. Malha retangular uniforme de quatro nós



Já a malha constituída de elementos triangulares de três nós, descrito na **SEÇÃO 1.9**, para o problema térmico e térmico estrutural, apresentou resultados satisfatórios, sendo utilizada neste trabalho. No total, malha possui 321 nós e 553 elementos, conforme a Figura 50.

Figura 50. Malha triangular regular uniforme de três nós



7. RESULTADOS

Os resultados do algoritmo considerando as temperaturas apresentadas na **Tabela 3**, foram comparados com os resultados obtidos no programa ANSYS® para validar o modelo de acoplamento proposto.

Considerando as condições de contorno adotadas para o verão, os resultados de campo de temperatura e deslocamento na horizontal são apresentados nas **Figura 51 e Figura 52**, **Figura 53 e Figura 54**.

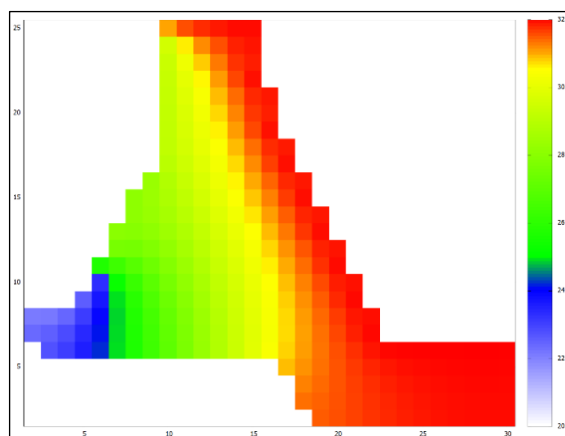


Figura 51. Campo de Temperatura (verão) do algoritmo

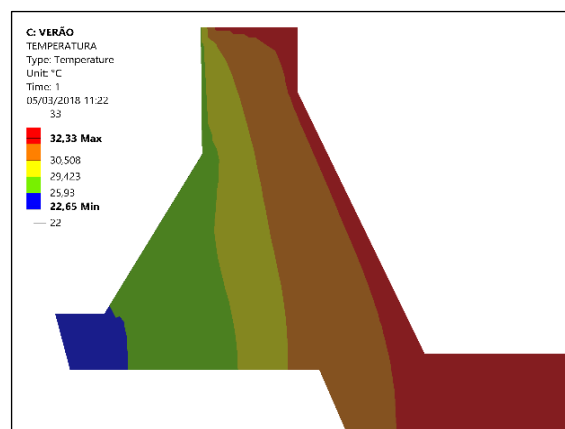


Figura 52. Campo de Temperatura (verão) do programa ANSYS®

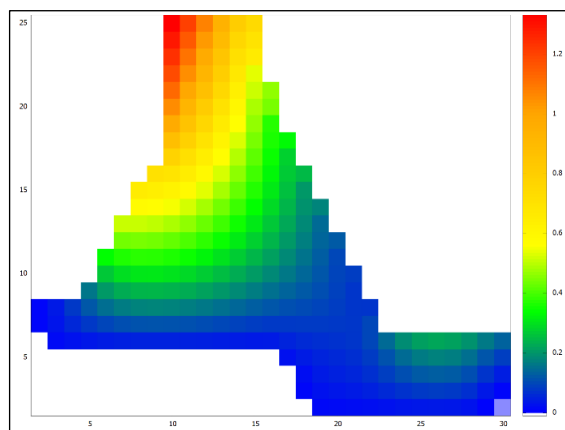


Figura 53. Campo de deslocamento na horizontal (verão) do algoritmo

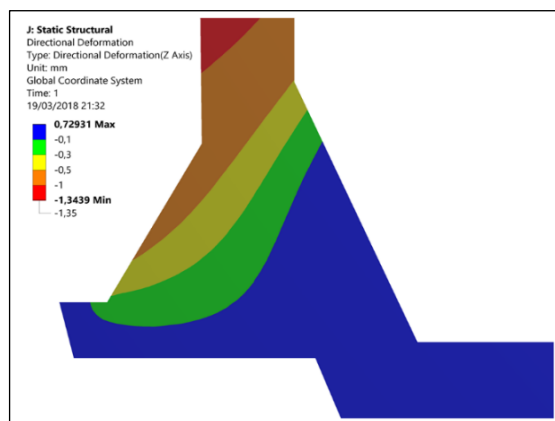


Figura 54. Campo de deslocamento na horizontal (verão) do programa ANSYS®

Considerando as condições de contorno considerado parao inverno, os resultados de campo de temperatura e deslocamento na horizontal são apresentados nas **Figura 55**, **Figura 56**, **Figura 57** e **Figura 58**.

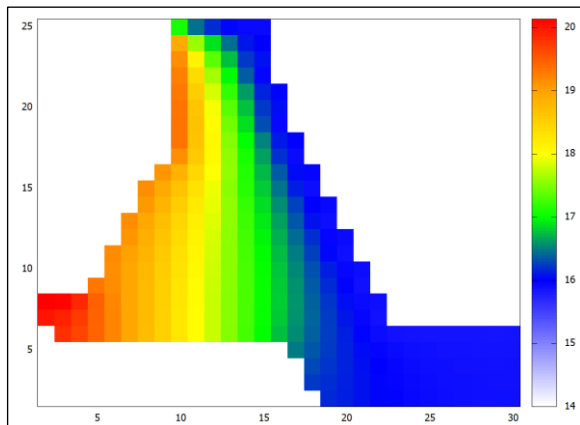


Figura 55. Campo de Temperatura do algoritmo

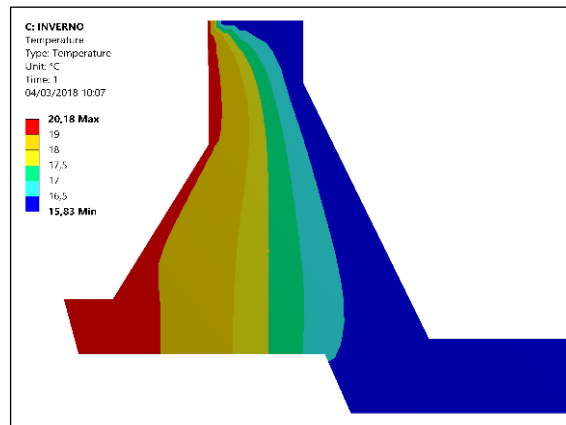


Figura 56. Campo de Temperatura do programa ANSYS®

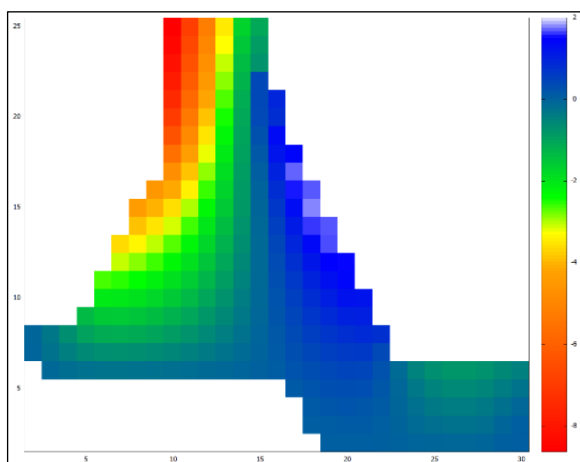


Figura 57. Campo de deslocamento na horizontal do algoritmo

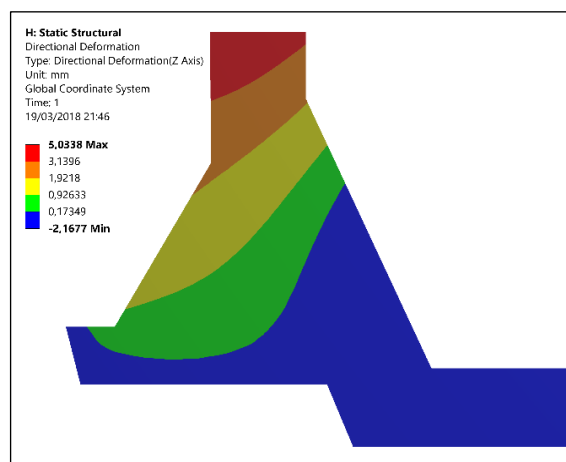


Figura 58. Campo de deslocamento na horizontal do programa ANSYS®

8. CONCLUSÕES

Como primeira conclusão, pode-se avaliar a concordância entre os resultados obtidos dos modelos, do código desenvolvido e do programa Ansys®.

Em relação ao problema térmico, a maior diferença de temperatura nodal encontrada entre o modelo resolvido pelo algoritmo e o modelo do programa ANSYS® é de aproximadamente 0,5°C.

Em relação aos deslocamentos da estrutura devido a variação de temperatura, os dois modelos apresentaram um comportamento físico coerente, correspondente ao registrado no instrumento pêndulo da barragem. Na condição de verão, a crista dos modelos se deslocaram para jusante e na condição de inverno, para montante.

No verão, o deslocamento da crista da barragem estimado pelo algoritmo foi de 1,2 mm, pelo modelo do Ansys® o deslocamento foi de 1,35 mm. No inverno, o deslocamento do modelo do algoritmo foi 8 mm, e o do Ansys® foi de 5,03 mm.

Em média, o deslocamento total, de jusante a montante no período de um ano, apontado pelos modelos foi de 8 mm. De acordo com os dados dos instrumentos, em 2010 o deslocamento real da barragem foi de 5 mm. Essa diferença pode ser devido a hipótese adotada na análise térmica de regime permanente, o que estima temperaturas no núcleo da barragem superiores ao real. Mesmo assim, os modelos apresentam deslocamentos horizontais máximos e mínimos, dentro dos valores previstos do projeto da barragem.

De acordo com os termômetros internos, o núcleo da estrutura apresenta temperaturas abaixo das temperaturas de superfície durante o verão, e temperaturas acima das temperaturas de superfície durante o inverno. Isso é um forte indicativo de um comportamento transiente de transferência de calor no interior da barragem. Uma nova análise de temperatura variando ao longo do tempo deve ser estudada para aperfeiçoamento e aprimoramento do algoritmo na resolução de estruturas de concreto massa.

REFERÊNCIAS BIBLIOGRÁFICAS

ALVES, L. M. **Métodos dos Elementos Finitos** - Apostila Organizada Das Aulas De Métodos Dos Elementos Finitos Do Curso De Doutorado. Universidade Federal Do Paraná. 2007. Disponível em: <<http://www.portalsaberlivre.com.br/manager/uploads/apostilas/1316558246.pdf>>. Acesso em: 5 set. 2018.

ARAÚJO, J. M. **Thermal Stresses In Reinforced Concrete Pile Caps**. Teoria e Prática na Engenharia Civil, n.22, p.1-14, Outubro, 2013. Disponível em: <http://www.editoradunas.com.br/revistatpec/Art1_N22.pdf>. Acessado em 4 set 2018.

ASGHAR, B. M. **Fundamental Finite Element Analysis and Applications**. Hoboken: John Wiley & Sons. 2005.

BENEVOLO, N. **Diretrizes Para o Projeto de Barragens de Concreto**. IX Seminário Nacional De Grandes Barragens, Rio de Janeiro. 1973. Disponível em: <http://www.cbdb.org.br/documentos/site/1973/Diretrizes_para_Projeto_Barragem_Concreto%20-%2025-09-07.pdf>. Acesso em: 4 set 2018.

BOTELHO, G. G., PITANGUEIRA, R. L. S., FONSECA, F. T., **Sistema Computacional Orientado a Objetos para Análises Acopladas Termo-Estruturais pelo Método dos Elementos Finitos**. CILAMCE 2015 – XXXVI Ibero-Latin American Congress on Computational Methods in Engineering – Rio de Janeiro, Rio de, Brazil. 2015. Disponível em: <https://www.insane.dees.ufmg.br/wpcontent/uploads/2017/03/Artigo_Flavio_2015_CILAMCE.pdf> . Acesso em: 11 set 2018.

BRAGA, W. F. **Transmissão de calor**. São Paulo: Pioneira Thomson Learning, 2004.

CASTILHO, E. M. **Análise térmica de barragens de betão durante a construção**. Dissertação (Mestrado em Estruturas) – Engenharia de Estruturas, Instituto superior Tecnico Lisboa. Lisbon: [s.n.]. 2013.

CEASB-PTI, **Centro de Estudos Avançados em Segurança de Barragem** – Parque Tecnológico Itaipu.

COELHO, N. A. **Métodos Analíticos e Numéricos Para o Estudo dos Efeitos Termomecânicos no Concreto Massa Orientados às Barragens de Gravidade**. Tese (Doutorado em Estruturas e Construção Civil). Publicação E.TD-007A/16, Departamento de Engenharia Civil e Ambiental. Universidade de Brasília. Brasília. 2016.

FUNAHASHI JÚNIOR, E. I.; KUPERMAN, S. C.; VICENTE, G. R.; FORNI, E. S.; LIMA, E. C. **Simulação De Tensões Térmicas Da Sala De Radioterapia Do Hospital Das**

Clínicas De Ribeirão Preto-SP. In: 52° Congresso Brasileiro do Concreto. Fortaleza: Ibracon, 2010.

ÇENGEL, Y. A.; GHAJAR, A. J.; **Transferência de Calor e Massa - Uma Abordagem Prática.** 4a Ed. MCGRAW-HILL: 2012.

HETNARSKI, R., ESLAMI, M., **Thermal Stresses – Advanced Theory and Applications.** Springer. 2009.

HICKMANN, T. **Análise Do Efeito da Variação Térmica Sazonal em Barragem de Contrafortes.** Tese (Doutorado em Métodos Numéricos em Engenharia) – Universidade Federal do Paraná, Curitiba. 2016.

INCROPERA, FRANK P.; DEWITT, DAVID P. **Fundamentos de Transferência de Calor e Massa.** 4ª Edição. LTC Editora. 1998.

ITAIPU BINACIONAL. **SAT-SISTEMA DE ARQUIVOS TÉCNICOS.**

JAAFAR, M. S., BAYAGOOB, K. H., NOORZAEI, J., THANOON, W. A. M. **Development of Finite Element Computer Code For Thermal Analysis of Roller Compacted Concrete Dams.** Advances in Engineering Software. Elsevier. 2007.

JAIME G. & MOLINA P. **Fundamentos del Método de Elemento Finito.** Primera Edición. Ingeniería Mecánica Universidad Mayor de San Andrés (UMSA). 2010.

JESUS, R. D. **Optimização da Forma Estrutural de uma Barragem.** Dissertação (Mestrado - Especialização em Estruturas) - Universidade do Porto. 2011.

KAVAMURA, E. E. **Estudo do comportamento termo-estrutural bidimensional de barragens de concreto utilizando o Método dos Elementos Finitos.** Dissertação (Mestrado em Métodos Numéricos em Engenharia) - Universidade Federal do Paraná, Curitiba. 2016.

KIM, N. H. SANKAR, B. V. **Introdução à Análise e ao Projeto em Elementos Finitos.** LTC Editora, 2011.

KLEMCZAK, B.; KNOPPIK, A. W.K.. **Reinforced Concrete Tank Walls and Bridge Abutments: Early-Age Behaviour, Analytic Approaches and Numerical Models.** Engineering Structures, February: 2015.

LÉGER, P., SEYDOU, S. **Seasonal Thermal Displacements of Gravity Dams Located in Northern Regions.** Journal of Performance of Constructed Facilities volume 23. 2009.

KHAN, M. I.; **Factors Affecting The Thermal Properties Of Concrete And Applicability Of Its Prediction Models.** Building and Environment. February. 2001.

LOGAN, D. L. **A First Course in the Finite Element Method**. Ed. CL-Engineering, 4th edition, 2006.

MIRZABOZORG, H. HARIRI-ARDEBILI, M.A. HESHMATI, M. SEYED-KOLBADI, S.M. **Structural Safety Evaluation Of Karun III Dam and Calibration of Finite Element Model Using Instrumentation and Site Observation**. Elsevier. Case Studies in Structural Engineering. Volume 1. 2014. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S2214399814000034>>. Acesso em 10 set 2018.

SA, E. C. E. B. **Critérios de projeto civil de usinas hidrelétricas**. Rio de Janeiro, 2003.

SOUZA, R. M. **O Método Dos Elementos Finitos Aplicado Ao Problema De Condução De Calor**. Universidade Federal Do Pará, Belém, 2003. Disponível em: <http://www.ufpa.br/nicae/integrantes/remo_souza/TrabPublicados/Apostilas/ApostilaElementosFinitosNiCAE.pdf>. Acesso em: 10 set 2018.

SILVA JUNIOR, E. J. ARACAYO, L. COELHO, D. **Comparação entre o método pseudo-dinâmico e o método espectro resposta aplicado como análise sísmica em uma barragem tipo: estudo de caso itaipu**. CILAMCE 2017 – XXXVIII Ibero-Latin American Congress on Computational Methods in Engineering - Florianópolis, Santa Catarina, Brazil. 2017. Disponível em: <http://www.swge.inf.br/PDF/CILAMCE2017-0645_077643.PDF>. Acesso em: 11 nov 2017.

TATIN, M., BRIFFAUT, M., DUFOUR, F., SIMON A., FABRE J. P. **Thermal Displacements Of Concrete Dams: Accounting For Water Temperature In Statistical Models**. Engineering Structures. VOLUME 91. EISEVIER. 2015). Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0141029615000681?via%3Dihub>>. Acesso em: 28 set. 2018.

USBR. **Design of Small Dams**. United States Department of the Interior Bureau of Reclamation. Denver. 1977.

VICENTE, G. R.; KUPERMAN, S. C.; FUNAHASHI JÚNIOR, E. I. **Fissuração de Origem Térmica em Blocos de Fundação: Quando Refrigerar o Concreto** In: 56 Congresso Brasileiro do Concreto. Natal: Ibracon: 2014.

ZHU BUFANG. **Thermal Stresses and Temperature Control of Mass Concrete**. Elsevier. First Edition. China Institute of Water Resources and Hydropower Research and Chinese Academy of Engineering. 2004.

BATHE, K.J. **Solution methods for large generalized eigenvalue problems**. Berkeley. Analyse dynamique et module par la m.e.f. LAN 189. Paris, 1977.

APÊNDICE I – ARQUIVO .TXT DE ENTRADA DE DADOS

ENTRADA_BLOCO_E6_THERMAL_STRESS_PLANE_STRAIN_SUMMER_11_01_2010

```

!*****
!ANÁLISE TÉRMOELÁSTICA BIDIMENSIONAL DA BARRAGEM DE CONTRAFORTE DE ITAIPU UTILISANDO MEF

!ALGORITMO DESENVOLVIDO DE ACORDO COM O PLANO DE TRABAHO APRESENTADO À BANCA
!EXAMINADORA DO CURSO DE ENGENHARIA CIVIL DE INFRAESTRUTURA DA UNILA, COMO
!PARTE DOS REQUISITOS PARA OBTENÇÃO DO GRAU DE BACHAREL EM ENGENHARIA CIVIL
!
!DISCENTE:   EDIVALDO JOSÉ DA SILVA JUNIOR
!ORIENTADOR: AREF KALILO LIMA KZAM
!
!PROGRAMA 2 - CÁLCULO DE TENSÕES TÉRMICAS EM FUNÇÃO DO CAMPO DE TEMPERATURA 2D

!*****
!MODELO: BLOCO E6 THERMAL STRESS PROBLEM

!*****

NODE_NUMBER, ELEM_NUMBER, MATERIAL_NUMBER, Elem.NBC_NUMBER, NODE.NBC_NUMBER, NODE_RESTRICTION_MOVE, STANDARD_TEMP
321, 553, 1, 0, 52, 32, 22
!*****
ELEMENT STANDARD SIZE X, Y - FOR GNUPLOT COLLOR PLOT GRAPHICS
4000.,4000.
!*****
NODE, X[m], Y[m] (COORDENADAS)
1, 36., 96.
2, 40., 96.
3, 44., 96.
4, 48., 96.
...
318, 104., 0.
319, 108., 0.
320, 112., 0.
321, 116., 0.
!*****
Material, E(mód.Elast)[GPa], v(coef.Poisson), k(Cond.Térmica), alpha(coef.dilat.Term.)[°C]
1, 35000., 0.16, 1.8492, 0.0001
!*****
ELEMENTO, PLANE STRESS[1]/STRAIN[2] ELEM_THICKNESS[m], Material, NÓ1, NÓ2, NÓ3
1, 2, 1., 1,          7, 2, 1
2, 2, 1., 1,          7, 8, 2
3, 2, 1., 1,          8, 3, 2
...
546, 2, 1., 1,          317, 306, 305
550, 2, 1., 1,          319, 308, 307
551, 2, 1., 1,          319, 320, 308
552, 2, 1., 1,          320, 309, 308
553, 2, 1., 1,          320, 321, 309

```

```

|*****
ELEMENTO, CONDIÇÃO NATURAL DE CONTORNO(NBC) NBC1, NBC2, NBC3 [0=não, 1=sim]
|*****
NÓ, FORÇA, DESLOC, FONTE_CALOR(Q), TEMPERATURA(T), FLUXO_CALOR(q), CONVECÇÃO(h) [0=não, 1=sim]
1, 0, 0, 0, 1, 0, 0
2, 0, 0, 0, 1, 0, 0
3, 0, 0, 0, 1, 0, 0
4, 0, 0, 0, 1, 0, 0
5, 0, 0, 0, 1, 0, 0
...
153, 0, 0, 0, 1, 0, 0
173, 0, 0, 0, 1, 0, 0
172, 0, 0, 0, 1, 0, 0
171, 0, 0, 0, 1, 0, 0
170, 0, 0, 0, 1, 0, 0
|*****
NÓ, FORÇA X, FORÇA Y, DESLOC X, DESLOC Y
1, 0., 0., 0., 0.
2, 0., 0., 0., 0.
3, 0., 0., 0., 0.
4, 0., 0., 0., 0.
5, 0., 0., 0., 0.
...
173, 0., 0., 0., 0.
172, 0., 0., 0., 0.
171, 0., 0., 0., 0.
170, 0., 0., 0., 0.
|*****
NÓ, FONTE_CALOR(Q), TEMPERATURA(T), FLUXO_CALOR_X(qx), FLUXO_CALOR_Y(qy), CONVECÇÃO_X(hX),
CONVECÇÃO_Y(hY)
1, 0., 32,23, 0., 0., 0., 0.
2, 0., 32,23, 0., 0., 0., 0.
3, 0., 32,23, 0., 0., 0., 0.
4, 0., 32,23, 0., 0., 0., 0.
...
137, 0., 22,65, 0., 0., 0., 0.
153, 0., 22,65, 0., 0., 0., 0.
173, 0., 22,65, 0., 0., 0., 0.
172, 0., 22,65, 0., 0., 0., 0.
171, 0., 22,65, 0., 0., 0., 0.
170, 0., 22,65, 0., 0., 0., 0.
|*****
NÓ, ID_RESTR_X, ID_RESTR_Y [0=LIVRE; 1=RESTRITO],
170, 1, 1
192, 1, 1
213, 1, 1
242, 1, 1
...
314, 1, 1
315, 1, 1
316, 1, 1
317, 1, 1
318, 1, 1
319, 1, 1
320, 1, 1
321, 1, 1

```

APÊNDICE II – ALGORITMO COMPLETO

```

|*****
!ANÁLISE TÉRMO-MECÂNICA 2D DA BARRAGEM DE CONTRAFORTE DA ITAIPU PELO
!MÉTODO DOS ELEMENTOS FINITOS EM FORTRAN

!ALGORITMO DESENVOLVIDO DE ACORDO COM O PLANO DE TRABAHO APRESENTADO À BANCA
!EXAMINADORA DO CURSO DE ENGENHARIA CIVIL DE INFRAESTRUTURA DA UNILA, COMO
!PARTE DOS REQUISITOS PARA OBTENÇÃO DO GRAU DE BACHAREL EM ENGENHARIA CIVIL
!
!DISCENTE:   EDIVALDO JOSÉ DA SILVA JUNIOR
!ORIENTADOR: AREF KALILO LIMA KZAM
!
!PROGRAMA - CÁLCULO CAMPO DE TEMPERATURA E DESLOCAMENTOS TÉRMICOS 2D
|*****

|***** PRÉ-PROCESSAMENTO *****
|*****
! Variáveis para alocar valores do TXT ****
|*****
program          THERMAL_STRESS_2D
implicit none

INTEGER(4)       i, j, k, m, n !CONTADORES
INTEGER          NODE_NUMBER,ELEM_NUMBER,MATERIAL_NUMBER,ELEM_NBC_NUMBER
INTEGER          NODE_RESTR_NUMBER, NODE_NBC_NUMBER, STANDARD_TEMP
REAL, ALLOCATABLE:: NODE_POSITION(:,), MATERIAL_PROP(:,)
INTEGER, ALLOCATABLE:: ELEM_MATERIAL(:,) , ID_STRESS_STRAIN(:,) , ELEM_NODE(:,) , ID_ELEM_NBC(:,)
INTEGER, ALLOCATABLE:: ID_NODE_NBC(:,)
REAL, ALLOCATABLE:: NBC_FORCA_X(:,) , NBC_FORCA_Y(:,) , NBC_DESLOC_X(:,) , NBC_DESLOC_Y(:,) , NBC_TEMP(:,)
REAL, ALLOCATABLE:: NBC_FONTE_CALOR(:,) , NBC_FLUXO_X(:,) , NBC_FLUXO_Y(:,) , NBC_CONVEC_X(:,) ,
NBC_CONVEC_Y(:,)
INTEGER, ALLOCATABLE:: NODE_RESTR(:,)

|*****
! PROCESSAMENTO - VARIÁVEIS PARA O CÁLCULO DO CAMPO DE TEMPERATURA 2D
|*****
REAL          a, b, ky, kx, k_coef
REAL, ALLOCATABLE:: L_CONDUCTION_MATRIX(:,,:), L_CONVECTION_MATRIX(:,,:), r_beta_t(:,), RQ_VETOR(:,)
REAL, ALLOCATABLE:: L_THERMAL_MATRIX(:,,:), r_ELEM_VECTOR(:,), G_THERMAL_MATRIX(:,)
REAL, ALLOCATABLE:: G_THERMAL_MATRIX_BACKUP(:,), V_GLOBAL(:,), VETOR_TEMP_NODE(:,)
VETOR_TEMP_NODE_mult(:)
REAL          ALPHA, BETA, beta_fluxo, beta_convect, convect_media, temp_media
REAL, ALLOCATABLE:: s(:,), t(:,), N_T(:,), VETOR_ELEM_TEMP(:,), CENTER_ELEM_TEMP(:,), VETOR_PIVOT(:,)
REAL          TIE1, TIE2, TIE3

```

```

!*****
!PROCESSAMENTO - VARIÁVEIS PARA O CÁLCULO DOS DESLOCAMENTOS E TENSÕES TÉRMICAS 2D
!*****
REAL, ALLOCATABLE:: F1(:),F2(:),F3(:),B1(:),B2(:),B3(:)
REAL, ALLOCATABLE:: C1(:),C2(:),C3(:),ELEMENT_AREA(:)
REAL          X1, X2, X3, Y1, Y2, Y3
REAL          MOD_ELAST, POISSON,AREA, hA, CONST1, CONST2, CONST3, const4
REAL          DIV_CONST1, aldel_t, CONST5, CONST6, CONST7, CONST8, CONST9
REAL, ALLOCATABLE:: B_MATRIX_TRANSP(:,:,:),B_MATRIX(:,:,:), C_MATRIX(:,:,:), ELEM_THICKNESS(:)

REAL, ALLOCATABLE:: LOCAL_TS_MATRIX(:,:,:), G_TS_MATRIX(:,:), G_TS_MATRIX_BACKUP(:,:)
REAL, ALLOCATABLE:: G_FORCE_VECTOR(:), L_re_VETOR(:,:), G_DISP_VECTOR(:), ELEM_STRESS(:,:)
INTEGER          LIN_L, LIN_G, COL_L, COL_G

REAL          dx1, dy1, dx2, dy2, dx3, dy3
REAL          ELEM_AVE_TEMP, DELTA_TEMP
REAL          AVE_DISP_ELEM, AVE_THERMAL_ELEM, alpha_delta_t
    
```

```

!*****
!PROCESSAMENTO - VARIÁVEIS PARA INVETER MATRIZ LU=F
!*****
REAL, ALLOCATABLE:: THERMAL_INDX(:), THERMAL_STRESS_INDX(:) !Variável utilizada pela biblioteca de resolução de
sistemas LU
REAL          ddd !Variável utilizada pela biblioteca de resolução de sistemas LU
    
```

```

!*****
!*** PÓS-PRECESSAMENTO - VARIÁVEIS PARA PLOTAR RESULTADOS ****
!*****
REAL          DISP_X_MAX, DISP_X_MIN, DISP_Y_MAX, DISP_Y_MIN
REAL          X_MAX, Y_MAX
REAL, ALLOCATABLE:: MATRIX_VISUAL_2D(:,:), MATRIX_VISUAL_2D_2(:,:)
INTEGER          X, Y
REAL          STAND_X, STAND_Y
    
```

```

!*****
! PRÉ-PROCESSAMENTO: LEITURA DE DADOS DO TXT
!*****
!open(unit=1,file="teste1_thermal_stress_Example 7_6_milímetros.txt")
!open(unit=1,file="ENTRADA1_EXEMPLO_VALIDACAO.txt")
open(unit=1,file="ENTRADA_BLOCO_E6_THERMAL_STRESS_PLANE_STRAIN_WINTER.txt")
!open(unit=1,file="ENTRADA_BLOCO_E6_THERMAL_STRESS_PLANE_STRAIN_SUMMER_11_01_2010.txt")

read(1,*) !ESPAÇO RESERVADO PARA O CABEÇALHO DO PROBLEMA NO TXT
read(1,*)
read(1,*) ! "
read(1,*) ! "
read(1,*) ! "
read(1,*) ! "
read(1,*) ! "
read(1,*) ! "
read(1,*) ! "
read(1,*) ! "
read(1,*) ! "
read(1,*) ! "
    
```



```
read(1,*) ! "  
read(1,*) ! "  
read(1,*) ! "  
read(1,*) ! "  
read(1,*) ! "  
read(1,*) ! "  
read(1,*) ! "  
read(1,*) ! "  
  
read(1,*)  
!*****  
write(*,*)"TOTAL NODAL, TOTAL ELEM, MATERIAL NUMBERS, ELEM.NBC NUMBER, NODE NBC NUMBER,  
NODE_RESTR_NUMBER, STANDARD TEMPERATURE"  
read(1,*) NODE_NUMBER,ELEM_NUMBER,MATERIAL_NUMBER,ELEM_NBC_NUMBER,  
NODE_NBC_NUMBER,NODE_RESTR_NUMBER, STANDARD_TEMP  
write(*,*) NODE_NUMBER,ELEM_NUMBER,MATERIAL_NUMBER,ELEM_NBC_NUMBER, NODE_NBC_NUMBER,  
NODE_RESTR_NUMBER, STANDARD_TEMP  
write(*,*)  
  
read(1,*)  
read(1,*)  
!*****  
STAND_X=0.D0; STAND_Y=0.D0  
  
!write(*,*)"ELEMENT STANDARD SIZE"  
read(1,*)STAND_X, STAND_Y  
!write(1,*)STAND_X, STAND_Y  
  
read(1,*)  
read(1,*)  
  
!*****  
allocate(NODE_POSITION(NODE_NUMBER ,2)); NODE_POSITION=0.D0  
!write(*,*)"NODE, X, Y (COORDENADAS)"  
  
do i=1, NODE_NUMBER  
read(1,*) n, (NODE_POSITION(n,j),j=1,2)  
!write(*,*) n, (NODE_POSITION(n,j),j=1,2)  
end do  
  
read(1,*)  
read(1,*)  
  
!NODE_POSITION=NODE_POSITION*1000! CONVERTENDO METROS EM MILÍMETROS!  
!*****  
allocate(MATERIAL_PROP(MATERIAL_NUMBER,4));MATERIAL_PROP=0.D0  
!write(*,*)  
!write(*,*)"MATERIAL, E(mod.Elast)[GPa], v(coef.Poisson), k(Cond.Termica), alpha(coef.dilat.Term.)[C]"  
  
do i=1, MATERIAL_NUMBER  
read(1,*) n,(MATERIAL_PROP(n,j),j=1,4)  
!write(*,*)n,(MATERIAL_PROP(n,j),j=1,4)  
end do  
  
read(1,*)  
read(1,*)
```

```

!*****
allocate(ELEM_MATERIAL(ELEM_NUMBER),ELEM_THICKNESS(ELEM_NUMBER));ELEM_MATERIAL=0.D0;
ELEM_THICKNESS=0.D0
allocate(ID_STRESS_STRAIN(ELEM_NUMBER), ELEM_NODE(ELEM_NUMBER,3));ID_STRESS_STRAIN=0.D0; ELEM_NODE=0.D0
!write(*,*)
!write(*,*)"ELEMENT, ID_STRESS_STRAIN, ELEM_THICKNESS"[m], PROP, Node 1, Node 2, Node 3"

do i=1, ELEM_NUMBER
read(1,*) n, ID_STRESS_STRAIN(n), ELEM_THICKNESS(n), ELEM_MATERIAL(n),(ELEM_NODE(n,i),j=1,3)
!write(*,*)n, ID_STRESS_STRAIN(n), ELEM_THICKNESS(n), ELEM_MATERIAL(n),(ELEM_NODE(n,i),j=1,3)
end do

read(1,*)
read(1,*)
!*****
allocate(ID_ELEM_NBC(ELEM_NUMBER,4)); ID_ELEM_NBC=0.D0

!write(*,*)
!write(*,*)"ELEMENT, CONDICAO NATURAL DE CONTORNO(NBC) NBC1, NBC2, NBC3 [0=NAO, 1=SIM]"

do i=1, ELEM_NBC_NUMBER
read(1,*) n, (ID_ELEM_NBC(n,i),j=1,3)
!write(*,*) n,(ID_ELEM_NBC(n,i),j=1,3)
end do

read(1,*)
read(1,*)
!*****
allocate(ID_NODE_NBC(NODE_NUMBER,6));ID_NODE_NBC=0.D0
!write(*,*)
!write(*,*)"NODE, FORCA, DESLOC, FONTE(Q), TEMPERATURA(T) FLUXO(q),CONVECÇÃO(h) [0=não, 1=sim]"

do i=1, NODE_NBC_NUMBER
read(1,*)n,(ID_NODE_NBC(n,i),j=1,6)
! write(*,*)n,(ID_NODE_NBC(n,i),j=1,6)
end do

read(1,*)
read(1,*)
!*****
allocate(NBC_FORCA_X(NODE_NUMBER), NBC_FORCA_Y(NODE_NUMBER));NBC_FORCA_X=0.D0;NBC_FORCA_Y=0.D0
allocate(NBC_DESLOC_X(NODE_NUMBER), NBC_DESLOC_Y(NODE_NUMBER));NBC_DESLOC_X=0.D0;NBC_DESLOC_Y=0.D0
allocate(NBC_FONTE_CALOR(NODE_NUMBER),NBC_TEMP(NODE_NUMBER)); NBC_FONTE_CALOR=0.d0; NBC_TEMP=0.D0
allocate(NBC_FLUXO_X(NODE_NUMBER), NBC_FLUXO_Y(NODE_NUMBER))
allocate(NBC_CONVEC_X(NODE_NUMBER), NBC_CONVEC_Y(NODE_NUMBER))
NBC_FLUXO_X=0.D0; NBC_FLUXO_Y=0.D0; NBC_CONVEC_X=0.D0; NBC_CONVEC_Y=0.D0
!write(*,*)
!write(*,*)"Node, FORCA X, FORCA Y, DESLOC X, DESLOC Y"

do j=1, NODE_NBC_NUMBER
read(1,*) n, NBC_FORCA_X(n), NBC_FORCA_Y(n), NBC_DESLOC_X(n), NBC_DESLOC_Y(n)
!write(*,*)n, NBC_FORCA_X(n), NBC_FORCA_Y(n), NBC_DESLOC_X(n), NBC_DESLOC_Y(n)
end do

read(1,*)
read(1,*)

```

```

!*****
!write(*,*)
!write(*,*)"Node, Q, T, qx, qy, hx, hy"

do j=1, NODE_NBC_NUMBER
  read(1,*) n, NBC_FONTE_CALOR(n),NBC_TEMP(n), NBC_FLUXO_X(n), NBC_FLUXO_Y(n), NBC_CONVEC_X(n),
NBC_CONVEC_Y(n)
  !write(*,*)n, NBC_FONTE_CALOR(n),NBC_TEMP(n), NBC_FLUXO_X(n), NBC_FLUXO_Y(n), NBC_CONVEC_X(n),
NBC_CONVEC_Y(n)
end do

  read(1,*)
  read(1,*)
!*****
allocate(NODE_RESTR(NODE_NUMBER, 2)); NODE_RESTR=0.D0
! write(*,*)"Node,RESTR X, RESTR Y"

do j=1, NODE_RESTR_NUMBER
  read(1,*) n, (NODE_RESTR(n,k),k=1,2)
  !write(*,*)n, (NODE_RESTR(n,k),k=1,2)
end do

CLOSE(1)

WRITE(*,*)">> PRE-PROCESSAMENTO OK"
!*****

!***** PROCESSAMENTO *****

NODE_POSITION=NODE_POSITION*1000 ! convertendo as coordenadas nodais m --> mm

!*****
!PARTE 1 - THERMAL
!*****

!*****
! COMPONENTES DA MATRIZ DE RIGIDEZ DO ELEMENTO
!*****
allocate(F1(ELEM_NUMBER),F2(ELEM_NUMBER),F3(ELEM_NUMBER)); F1=0.D0;F2=0.D0;F3=0.D0
allocate(B1(ELEM_NUMBER),B2(ELEM_NUMBER),B3(ELEM_NUMBER)); B1=0.D0;B2=0.D0;B3=0.D0
allocate(C1(ELEM_NUMBER),C2(ELEM_NUMBER),C3(ELEM_NUMBER)); C1=0.D0;C2=0.D0;C3=0.D0
allocate(ELEMENT_AREA(ELEM_NUMBER)); ELEMENT_AREA=0.D0

allocate(L_CONDUCTION_MATRIX(ELEM_NUMBER,3,3));L_THERMAL_MATRIX=0.D0
do i=1, ELEM_NUMBER

  X1=NODE_POSITION(ELEM_NODE(i,1),1)
  X2=NODE_POSITION(ELEM_NODE(i,2),1)
  X3=NODE_POSITION(ELEM_NODE(i,3),1)

  Y1=NODE_POSITION(ELEM_NODE(i,1),2)
  Y2=NODE_POSITION(ELEM_NODE(i,2),2)
  Y3=NODE_POSITION(ELEM_NODE(i,3),2)

```

```

F1(i)= X2*Y3-X3*Y2
F2(i)= X3*Y1-X1*Y3
F3(i)= X1*Y2-X2*Y1

B1(i)= Y2-Y3
B2(i)= Y3-Y1
B3(i)= Y1-Y2

C1(i)= X3-X2
C2(i)= X1-X3
C3(i)= X2-X1

ELEMENT_AREA(i) = (F1(i)+F2(i)+F3(i))/2

!*****
! MATRIZ DE RIGIDEZ DO ELEMENTO
!*****
AREA=ELEMENT_AREA(i)
k_coef=MATERIAL_PROP(ELEM_MATERIAL(i),3)

L_CONDUCTION_MATRIX(i,1,1)= k_coef*(B1(i)**2+C1(i)**2)*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,1,2)= k_coef*(B1(i)*B2(i)+C1(i)*C2(i))*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,1,3)= k_coef*(B1(i)*B3(i)+C1(i)*C3(i))*(1/(4*AREA))

L_CONDUCTION_MATRIX(i,2,1)= k_coef*(B1(i)*B2(i)+C1(i)*C2(i))*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,2,2)= k_coef*(B2(i)**2+C2(i)**2)*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,2,3)= k_coef*(B2(i)*B3(i)+C2(i)*C3(i))*(1/(4*AREA))

L_CONDUCTION_MATRIX(i,3,1)= k_coef*(B1(i)*B3(i)+C1(i)*C3(i))*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,3,2)= k_coef*(B2(i)*B3(i)+C2(i)*C3(i))*(1/(4*AREA))
L_CONDUCTION_MATRIX(i,3,3)= k_coef*(B3(i)**2+C3(i)**2)*(1/(4*AREA))

end do

!*****
!REALIZAR AS SOMAS (Kk+Kp+K_alpha)d=rq+r_beta (PROBLEMAS QUE ENVOLVEM CONVECÇÃO E FONTE DE CALOR)
!*****
allocate(L_THERMAL_MATRIX(ELEM_NUMBER,3,3), r_ELEM_VECTOR(ELEM_NUMBER,3)); L_THERMAL_MATRIX=0.D0;
r_ELEM_VECTOR=0.D0

do i=1, ELEM_NUMBER
do j=1,3
! r_ELEM_VECTOR(i,j) = RQ_VETOR(i,j)+r_beta_t(i,j) !FONTE DE CALOR

do k=1,3

L_THERMAL_MATRIX(i,j,k) = L_THERMAL_MATRIX(i,j,k) + L_CONDUCTION_MATRIX(i,j,k) ! + K_alpha(i,j,k)

end do
end do
end do

```

```

!*****
!*****MATRIZ GLOBAL DE RIGIDEZ *****
!*****
allocate(G_THERMAL_MATRIX(NODE_NUMBER,NODE_NUMBER), V_GLOBAL(NODE_NUMBER)); G_THERMAL_MATRIX=0.D0;
V_GLOBAL=0.D0
allocate(G_THERMAL_MATRIX_BACKUP(NODE_NUMBER,NODE_NUMBER)); G_THERMAL_MATRIX_BACKUP=0.D0

DO I=1, ELEM_NUMBER
do j=1,3

    V_GLOBAL(ELEM_NODE(i,j))= V_GLOBAL(ELEM_NODE(i,j)) !+ r_ELEM_VECTOR(i,j) !FONTE DE CALOR

do k=1, 3

G_THERMAL_MATRIX(ELEM_NODE(i,j),ELEM_NODE(i,k))=G_THERMAL_MATRIX(ELEM_NODE(i,j),ELEM_NODE(i,k))+L_THERMAL
_MATRIX(i,j,k)
    G_THERMAL_MATRIX_BACKUP(ELEM_NODE(i,j),ELEM_NODE(i,k))=G_THERMAL_MATRIX(ELEM_NODE(i,j),ELEM_NODE(i,k))

end do
end do
END DO

!*****
!*****APLICAR TEMPERATURAS DE CONTORNO NO VETOR GLOBAL DE TEMPERATURAS
!*****
ALLOCATE(VETOR_TEMP_NODE(NODE_NUMBER));VETOR_TEMP_NODE=0.D0

do i=1, NODE_NUMBER
    if(ID_NODE_NBC(i,4)==1)then
        VETOR_TEMP_NODE(i)=NBC_TEMP(i)
    endif
end do

!*****
!VISUALIZAR SE AS CONDIÇÕES FORAM ALOCADAS CORRETAMENTE

! WRITE(*,*)"*****"
! WRITE(*,*)"MATRIZ GLOBAL DE RIGIDEZ"
! CALL MATRIXPRINT (G_THERMAL_MATRIX, NODE_NUMBER, NODE_NUMBER)

! WRITE(*,*)"*****"
! WRITE(*,*)"VETOR DE FLUXO"
! do i=1,NODE_NUMBER
! WRITE(*,*) i, V_GLOBAL(i)
! end do

! WRITE(*,*)"*****"
! WRITE(*,*)"CONDICAO DE CONTORNO: TEMPERATURA"
! do i=1,NODE_NUMBER
! WRITE(*,*) i, VETOR_TEMP_NODE(i)
! end do

```

```

!*****
! DELETAR TODAS AS LINHAS QUE REPRESENTAM OS NÓS QUE POSSUEM NBC DE TEMPERATURA
!*****
do i=1, NODE_NUMBER
  IF(ID_NODE_NBC(i,4)==1)THEN
    G_THERMAL_MATRIX(i,:)=0.D0
    G_THERMAL_MATRIX(i,i)=1.D0
  END IF
END DO
!*****
! MULTIPLICAR AS COLUNAS DOS NÓS COM NBC DE TEMPERATURA PELO VETOR NBC_TEMP
!*****!
allocate(VETOR_TEMP_NODE_mult(NODE_NUMBER));VETOR_TEMP_NODE_mult=1.D0
allocate(VETOR_PIVOT(NODE_NUMBER));VETOR_PIVOT=0.D0
do i=1, NODE_NUMBER
  if(ID_NODE_NBC(i,4)==1)then
    VETOR_TEMP_NODE_mult(i)=VETOR_TEMP_NODE(i)
  end if
end do
!*****
! SOMAR O (SOMATÓRIO DAS COLULAS * NBC TEMP) DE CADA NÓ COM O VETOR DE FLUXO. INVERTER SINAL!!!
!*****
do i=1, NODE_NUMBER
do j=1, NODE_NUMBER
  IF(ID_NODE_NBC(i,4)==1)THEN
    VETOR_PIVOT(j)= VETOR_PIVOT(j)+G_THERMAL_MATRIX(j,i)*VETOR_TEMP_NODE_mult(i)
  END IF
END DO
END DO

V_GLOBAL=V_GLOBAL+(-vetor_PIVOT) !AQUI INVERTE SINAL

do i=1, NODE_NUMBER
  if(ID_NODE_NBC(i,4)==1)then
    V_GLOBAL(i)=0.D0
  endif
end do

!*****
!VISUALIZAR SE AS CONDIÇÕES FORAM ALOCADAS CORRETAMENTE
!WRITE(*,*)"*****"
! WRITE(*,*)"MATRIZ GLOBAL DE RIGIDEZ APOS CONDICoes DE CONTORNO"
! CALL MATRIXPRINT (G_THERMAL_MATRIX, NODE_NUMBER, NODE_NUMBER)
!WRITE(*,*)"*****"
!WRITE(*,*)"vetor de temperatura de multiplicacao"
! do i=1,NODE_NUMBER
! WRITE(*,*) i, VETOR_TEMP_NODE_mult(i)
! end do

!WRITE(*,*)"*****"
!WRITE(*,*)"VETOR DE SOMA DAS COLUNAS EXTRAIDAS"
! do i=1,NODE_NUMBER
! WRITE(*,*) i, VETOR_PIVOT(i)
! end do

```

```
!WRITE(*,*)"*****"
!WRITE(*,*)"VETOR Global de fluxo"
! do i=1,NODE_NUMBER
! WRITE(*,*) i, V_GLOBAL(i)
! end do
!*****

!*****
!SUBROTINAS QUE RESOLVEM O SISTEMA LU
!*****
allocate (THERMAL_INDX(NODE_NUMBER))
call ludcmp(G_THERMAL_MATRIX,NODE_NUMBER,NODE_NUMBER,THERMAL_INDX,ddd)
call lubksb(G_THERMAL_MATRIX,NODE_NUMBER,NODE_NUMBER,THERMAL_INDX,V_GLOBAL)
!*****

!*****
! RESULTADO DAS TEMPERATURAS NODAIS
!*****
VETOR_TEMP_NODE=VETOR_TEMP_NODE+V_GLOBAL

!WRITE(*,*)"*****"
! WRITE(*,*)"RESULTADO:"
! WRITE(*,*)"NODE, TEMPERATURE"
! do i=1,NODE_NUMBER
! WRITE(*,*) i, VETOR_TEMP_NODE(i)
! end do

!*****
WRITE(*,*)">> PROCESSAMENTO PARTE 1 - THERMAL OK"

!PROCESSAMENETO
!*****
!PART 2 - THERMAL STRESS
!*****

!*****
! COMPONENTES DA MATRIZ DE RIGIDEZ DO ELEMENTO
!*****
allocate(LOCAL_TS_MATRIX(ELEM_NUMBER,6,6));LOCAL_TS_MATRIX=0.D0
allocate(L_re_VETOR(ELEM_NUMBER,6));L_re_VETOR=0.D0

do i=1, ELEM_NUMBER

MOD_ELAST=MATERIAL_PROP(ELEM_MATERIAL(i),1)
POISSON=MATERIAL_PROP(ELEM_MATERIAL(i),2)

!*****
! MATRIZ DE RIGIDEZ DO ELEMENTO
!*****
AREA=ELEMENT_AREA(i)

if(ID_STRESS_STRAIN(i)==1)then
!*****
```

```

! ESTADO PLANO DE TENSÃO !!!!!
!*****
CONST1=(MOD_ELAST/(1-POISSON**2))
CONST2=(MOD_ELAST*POISSON/(1-POISSON**2))
CONST3=(MOD_ELAST*(1-POISSON)/(2*(1-POISSON**2)))
DIV_CONST1=(4*AREA**2)
ha=ELEM_THICKNESS(i)*ELEMENT_AREA(i)
CONST4 = MATERIAL_PROP(ELEM_MATERIAL(i),4)

else
!*****
! ESTADO PLANO DE DEFORMAÇÃO !!!!!
!*****
CONST1=(MOD_ELAST*(1-POISSON)/((1+POISSON)*(1-2*POISSON)))
CONST2=(MOD_ELAST*POISSON/((1+POISSON)*(1-2*POISSON)))
CONST3=(MOD_ELAST*(1-2*POISSON)/(2*(1+POISSON)*(1-2*POISSON)))
DIV_CONST1=(4*AREA**2)
ha=ELEM_THICKNESS(i)*ELEMENT_AREA(i)
CONST4 = (1+POISSON)*MATERIAL_PROP(ELEM_MATERIAL(i),4)

end if

LOCAL_TS_MATRIX(i,1,1)= (B1(i)**2*CONST1/DIV_CONST1 + C1(i)**2*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,1,2)= ((C1(i)*B1(i))/DIV_CONST1*(CONST2 + CONST3))*ha
LOCAL_TS_MATRIX(i,1,3)= (B2(i)*B1(i)*CONST1/DIV_CONST1 + C2(i)*C1(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,1,4)= (B1(i)*C2(i)*CONST2/DIV_CONST1 + B2(i)*C1(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,1,5)= (B3(i)*B1(i)*CONST1/DIV_CONST1 + C3(i)*C1(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,1,6)= (C3(i)*B1(i)*CONST2/DIV_CONST1 + B3(i)*C1(i)*CONST3/DIV_CONST1)*ha

LOCAL_TS_MATRIX(i,2,1)= ((C1(i)*B1(i))/DIV_CONST1*(CONST2 + CONST3))*ha
LOCAL_TS_MATRIX(i,2,2)= (C1(i)**2*CONST1/DIV_CONST1 + B1(i)**2*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,2,3)= (B2(i)*C1(i)*CONST2/DIV_CONST1 + C2(i)*B1(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,2,4)= (C1(i)*C2(i)*CONST1/DIV_CONST1 + B1(i)*B2(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,2,5)= (C1(i)*B3(i)*CONST2/DIV_CONST1 + B1(i)*C3(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,2,6)= (C1(i)*C3(i)*CONST1/DIV_CONST1 + B1(i)*B3(i)*CONST3/DIV_CONST1)*ha

LOCAL_TS_MATRIX(i,3,1)= (B1(i)*B2(i)*CONST1/DIV_CONST1 + C2(i)*C1(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,3,2)= (B2(i)*C1(i)*CONST2/DIV_CONST1 + B1(i)*C2(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,3,3)= (B2(i)**2*CONST1/DIV_CONST1 + C2(i)**2*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,3,4)= (B2(i)*C2(i)*CONST2/DIV_CONST1 + B2(i)*C2(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,3,5)= (B2(i)*B3(i)*CONST1/DIV_CONST1 + C3(i)*C2(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,3,6)= (B2(i)*C3(i)*CONST2/DIV_CONST1 + B3(i)*C2(i)*CONST3/DIV_CONST1)*ha

LOCAL_TS_MATRIX(i,4,1)= (B1(i)*C2(i)*CONST2/DIV_CONST1 + B2(i)*C1(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,4,2)= (C1(i)*C2(i)*CONST1/DIV_CONST1 + B2(i)*B1(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,4,3)= (C2(i)*B2(i)/DIV_CONST1*(CONST2 + CONST3))*ha
LOCAL_TS_MATRIX(i,4,4)= (C2(i)**2*CONST1/DIV_CONST1 + B2(i)**2*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,4,5)= (B3(i)*C2(i)*CONST2/DIV_CONST1 + B2(i)*C3(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,4,6)= (C3(i)*C2(i)*CONST1/DIV_CONST1 + B2(i)*B3(i)*CONST3/DIV_CONST1)*ha

LOCAL_TS_MATRIX(i,5,1)= (B1(i)*B3(i)*CONST1/DIV_CONST1 + C1(i)*C3(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,5,2)= (C1(i)*B3(i)*CONST2/DIV_CONST1 + C3(i)*B1(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,5,3)= (B2(i)*B3(i)*CONST1/DIV_CONST1 + C3(i)*C2(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,5,4)= (C2(i)*B3(i)*CONST2/DIV_CONST1 + B2(i)*C3(i)*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,5,5)= (B3(i)**2*CONST1/DIV_CONST1 + C3(i)**2*CONST3/DIV_CONST1)*ha
LOCAL_TS_MATRIX(i,5,6)= (B3(i)*C3(i)/DIV_CONST1*(CONST2 + CONST3))*ha

```



```

LOCAL_TS_MATRIX(i,6,1)= ((B1(i)*C3(i)*CONST2)/DIV_CONST1 + (B3(i)*C1(i)*CONST3)/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,6,2)= ((C1(i)*C3(i)*CONST1)/DIV_CONST1 + (B1(i)*B3(i)*CONST3)/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,6,3)= ((B2(i)*C3(i)*CONST2)/DIV_CONST1 + (C2(i)*B3(i)*CONST3)/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,6,4)= ((C2(i)*C3(i)*CONST1)/DIV_CONST1 + (B2(i)*B3(i))*CONST3/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,6,5)= (((B3(i)*C3(i))*(CONST2 + CONST3))/DIV_CONST1)*hA
LOCAL_TS_MATRIX(i,6,6)= ((C3(i)**2*CONST1)/DIV_CONST1 + (B3(i)**2*CONST3)/DIV_CONST1)*hA

!*****
! CONTRIBUIÇÃO DA TEMPERATURA --> The equivalent load vector from initial strains due to temperature change is
!*****
! AS TEMPERATURAS NODAIS SÃO UTILIZADAS COMO CONDIÇÃO DE FORÇA INICIAL DEVIDO A DILATAÇÃO TÉRMICA.
! VARIÁVEL DA TEMPERATURA NODAL VETOR_TEMP_NODE(ELEM_NODE(i,J)

L_re_VETOR(i,1)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,1))-
STANDARD_TEMP)*B1(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))
L_re_VETOR(i,2)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,1))-
STANDARD_TEMP)*C1(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))
L_re_VETOR(i,3)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,2))-
STANDARD_TEMP)*B2(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))
L_re_VETOR(i,4)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,2))-
STANDARD_TEMP)*C2(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))
L_re_VETOR(i,5)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,3))-
STANDARD_TEMP)*B3(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))
L_re_VETOR(i,6)=const4*(VETOR_TEMP_NODE(ELEM_NODE(i,3))-
STANDARD_TEMP)*C3(i)*(CONST1+CONST2)*hA/(2*ELEMENT_AREA(i))
!*****
end do

!*****
!CALCULA MATRIZ GLOBAL
!*****
ALLOCATE (G_TS_MATRIX(2*NODE_NUMBER, 2*NODE_NUMBER)); G_TS_MATRIX=0.D0
ALLOCATE (G_TS_MATRIX_BACKUP(2*NODE_NUMBER, 2*NODE_NUMBER)); G_TS_MATRIX=0.D0

DO I=1, ELEM_NUMBER
DO J=1, 3 !NUMERO DE NOS DO ELEMENTO
DO K=1, 2 ! NUMERO DE GRAUS DE LIBERDADE DO NO

DO M=1,3 !NUMERO DE NOS DO ELEMENTO
DO N= 1,2 !NUMERO DE GRAUS DE LIBERDADE

LIN_G=2*(ELEM_NODE(i,j)-1)+K
COL_G=2*(ELEM_NODE(i,M)-1)+N

LIN_L=2*(j-1)+k
COL_L= 2*(M-1)+N

G_TS_MATRIX(LIN_G,COL_G)=G_TS_MATRIX(LIN_G,COL_G)+LOCAL_TS_MATRIX(i,LIN_L,COL_L )
G_TS_MATRIX_BACKUP(LIN_G,COL_G)=G_TS_MATRIX(LIN_G,COL_G)

END DO
END DO
END DO
END DO
END DO

```

```

!*****
!aplicar das restricoes na matriz global
!*****
DO i=1, NODE_NUMBER
do j=1, 2
  IF(NODE_RESTR(i,j)==1) THEN
  if(j==1)G_TS_MATRIX(2*(i-1)+1,:)=0.D0
  if(j==1)G_TS_MATRIX(:,2*(i-1)+1)=0.D0
  if(j==1)G_TS_MATRIX(2*(i-1)+1,2*(i-1)+1)=1.0D0

  if(j==2)G_TS_MATRIX(2*(i-1)+2,:)=0.D0
  if(j==2)G_TS_MATRIX(:,2*(i-1)+2)=0.D0
  if(j==2)G_TS_MATRIX(2*(i-1)+2,2*(i-1)+2)=1.0D0
  END IF
END DO
end do

!*****
!VISUALIZAR SE AS CONDIÇÕES FORAM ALOCADAS CORRETAMENTE

!WRITE(*,*)
!WRITE(*,*)'*****'
!write(*,*)'MATRI DE RIGIDEZ GLOBAL'
!CALL MATRIXPRINT (G_TS_MATRIX_BACKUP, 2*NODE_NUMBER, 2*NODE_NUMBER)

!WRITE(*,*)
!WRITE(*,*)'*****'
!WRITE(*,*)'APOS RESTRICOES'

!CALL MATRIXPRINT (G_TS_MATRIX, 2*NODE_NUMBER, 2*NODE_NUMBER)
!*****

!*****
!VETOR GLOBAL DE FORÇAS
!*****
!SOMATÓRIO DE FORÇAS NODAIS, PRESSÃO NO ELEMENETO E FORÇA DEVIDO DILATAÇÃO TÉRMICA

ALLOCATE(G_FORCE_VECTOR(2*NODE_NUMBER));G_FORCE_VECTOR=0.D0
ALLOCATE(G_DISP_VECTOR(2*NODE_NUMBER));G_DISP_VECTOR=0.D0

DO I=1,ELEM_NUMBER
n=1

do j=1,3
do k=1,2

G_FORCE_VECTOR(2*(ELEM_NODE(i,j)-1)+k)= L_re_VETOR(i,n) + G_FORCE_VECTOR(2*(ELEM_NODE(i,j)-1)+k)

n=n+1

END DO
end do
end do

```

```

!WRITE(*,*)'VETOR GLOBAL DE FORÇAS TERMICAS'
!do i=1, 2*node_number
!  write(*,*)i, G_FORCE_VECTOR(i)
!end do

!PREPARANDO O VETOR DE DESLOCAMENTO PARA A SUBROTINA QUE INVERTE MATRIZ

G_DISP_VECTOR=G_FORCE_VECTOR

DO i=1, NODE_NUMBER
  do j=1, 2
    IF(NODE_RESTR(i,j)==1) THEN

      if(j==1)G_DISP_VECTOR(2*(i-1)+1)=0.D0 !APLICANDO RESTRIÇÃO
      if(j==2)G_DISP_VECTOR(2*(i-1)+2)=0.D0 !APLICANDO RESTRIÇÃO
      END IF
    end do
  end do

!*****
WRITE(*,*)
WRITE(*,*)'VETOR GLOBAL DE FORÇAS TERMICAS APOS RESTRICAO'

do i=1, 2*node_number
  write(*,*)i, G_DISP_VECTOR(i)
end do
!*****

!*****
!SUBROTINAS QUE RESOLVEM O SISTEMA LU
!*****
allocate (THERMAL_STRESS_INDX(2*NODE_NUMBER))

call ludcmp(G_TS_MATRIX,2*NODE_NUMBER,2*NODE_NUMBER,THERMAL_STRESS_INDX,ddd)
call lubksb(G_TS_MATRIX,2*NODE_NUMBER,2*NODE_NUMBER,THERMAL_STRESS_INDX,G_DISP_VECTOR)

!*****
!VISUALIZAR RESULTADOS

!WRITE(*,*)
WRITE(*,*)"RESULTADO:"
WRITE(*,*)' NODE, NODE DISPLACEMENT:'

do i=1, 2*node_number
write(*,*)i, G_DISP_VECTOR(i)
end do

WRITE(*,*)">> PROCESSAMENTO PARTE 2 - THERMAL STRESS OK"
!*****

```

```

!***** POS-PROCESSAMENTO *****
if(STAND_X==0)then

  WRITE(*,*)>> POS-PROCESSAMENTO OK "

  WRITE(*,*)
  WRITE(*,*)"NAO HA MATRIZ 2D DE VISUALIZAÇÃO PORQUE OS ELEMENTOS NAO SAO UNIFORMES! =(

ELSE

!*****
!ROTINA PARA GERAR GRÁFICOS COLORIDOS NO GNOPLT
!*****
X_MAX=0.D0;Y_MAX=0.D0

do i=1, NODE_NUMBER
  if(X_MAX < NODE_POSITION(i,1))X_MAX=NODE_POSITION(i,1)
  if(Y_MAX < NODE_POSITION(i,2))Y_MAX=NODE_POSITION(i,2)
end do

X_MAX= X_MAX/STAND_X
Y_MAX= Y_MAX/STAND_Y

!*****
!GERANDO GRÁFICOS - CAMPO DE TEMPERATURA
!*****
!allocate(MATRIX_VISUAL_2D(X_MAX,Y_MAX)); MATRIX_VISUAL_2D=100.0d0 !!!!! !EXEMPLO DE VALIDAÇÃO
allocate(MATRIX_VISUAL_2D(X_MAX,Y_MAX)); MATRIX_VISUAL_2D=14.0d0 !!!!! !BLOCO E6 INVERNO
!allocate(MATRIX_VISUAL_2D(X_MAX,Y_MAX)); MATRIX_VISUAL_2D=20.0d0 !!!!! !BLOCO E6 VERÃO

do i=1, ELEM_NUMBER

  X= NODE_POSITION(ELEM_NODE(i,3),1)/STAND_X
  Y= NODE_POSITION(ELEM_NODE(i,3),2)/STAND_Y

  AVE_THERMAL_ELEM=(VETOR_TEMP_NODE(ELEM_NODE(i,1))+VETOR_TEMP_NODE(ELEM_NODE(i,2))+VETOR_TEMP_NODE
(ELEM_NODE(i,3)))/3

  MATRIX_VISUAL_2D(X,Y)=AVE_THERMAL_ELEM

end do

!*****
!SAIDA_MATRIX_GRAPHIC.TXT - CADA POSIÇÃO MATRICIAL REPRESENTA UM ELEMENTO
!*****
open(unit=2,file="SAIDA1_THERMAL_GRAPHIC_MATRIX.txt")
do i=1, Y_MAX
  write(2,*)(MATRIX_VISUAL_2D(j,i),j=1,X_MAX)
  !write(*,*)(MATRIX_VISUAL_2D(i,j),j=1,X_MAX)
end do

close(2)
!*****

```

```

!*****
!GERANDO GRÁFICOS - CAMPO DE DESLOCAMENTOS EM X
!*****
MATRIX_VISUAL_2D=0.0005D0 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!INVERNO
!MATRIX_VISUAL_2D=-0.0005D0 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!VERÃO
!MATRIX_VISUAL_2D=0.d0 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!EXEMPLO DE VALIDAÇÃO

do i=1, ELEM_NUMBER
  X= NODE_POSITION(ELEM_NODE(i,3),1)/STAND_X
  Y= NODE_POSITION(ELEM_NODE(i,3),2)/STAND_Y

  AVE_DISP_ELEM=(G_DISP_VECTOR(2*(ELEM_NODE(i,1)-1)+1)+G_DISP_VECTOR(2*(ELEM_NODE(i,2)-1)+1)+G_DISP_VECTOR(2*(ELEM_NODE(i,3)-1)+1))/3

  MATRIX_VISUAL_2D(X,Y)= AVE_DISP_ELEM

END DO
!*****
!SAIDA_MATRIX_GRAPHIC.TXT - CADA POSIÇÃO MATRICIAL REPRESENTA UM ELEMENTO
!*****
open(unit=3,file="SAIDA2_X_TS_DISP_GRAPHIC_MATRIX.txt")
do i=1, Y_MAX
  write(3,*)(MATRIX_VISUAL_2D(j,i),j=1,X_MAX)
end do

close(3)

!*****
!GERANDO GRÁFICOS - CAMPO DE DESLOCAMENTOS EM Y
!*****
MATRIX_VISUAL_2D=2.0D0 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!INVERNO
!MATRIX_VISUAL_2D=-0.01D0 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!VERÃO
!MATRIX_VISUAL_2D=0.d0 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!EXEMPLO DE VALIDAÇÃO

do i=1, ELEM_NUMBER
  X= NODE_POSITION(ELEM_NODE(i,3),1)/STAND_X
  Y= NODE_POSITION(ELEM_NODE(i,3),2)/STAND_Y

  AVE_DISP_ELEM=(G_DISP_VECTOR(2*(ELEM_NODE(i,1)-1)+2)+G_DISP_VECTOR(2*(ELEM_NODE(i,2)-1)+2)+G_DISP_VECTOR(2*(ELEM_NODE(i,3)-1)+2))/3

  MATRIX_VISUAL_2D(X,Y)= AVE_DISP_ELEM

END DO
!*****
!SAIDA_MATRIX_GRAPHIC.TXT - CADA POSIÇÃO MATRICIAL REPRESENTA UM ELEMENTO
!*****
open(unit=4,file="SAIDA3_Y_TS_DISP_GRAPHIC_MATRIX.txt")
do i=1, Y_MAX
  write(4,*)(MATRIX_VISUAL_2D(j,i),j=1,X_MAX)
end do

close(4)
!*****
WRITE(*,*)>> POS-PROCESSAMENTO OK"
WRITE(*,*)>UTILIZE O GNUPLOT PARA VISUALIZAR GRAFICAMENTE OS RESULTADOS =D"

end if
END PROGRAM

```

```
! EXIBIR MATRIZES DE FORMA ORGANIZADA NO TERMINAL
```

```
subroutine MATRIXPRINT (a, m, n)
```

```
integer m, n, i, j, jref
```

```
real a(m,n)
```

```
!
```

```
do 2000 jref = 0, n-1, 6
```

```
print '(2X,6I12)', (j, j=jref+1, min(jref+6, n))
```

```
do 1500 i = 1, m
```

```
print '(I5,1P6E12.2)', i, (a(i, j), j=jref+1, min(jref+6, n))
```

```
1500 continue
```

```
2000 continue
```

```
!
```

```
return
```

```
end
```

```
SUBROUTINE ludcmp(a,n,np,indx,d)
```

```
integer n,np,indx(n),nmax
```

```
real d,a(np,np),TINY
```

```
parameter (nmax=640,Tiny=1.0e-20)
```

```
integer i,imax,j,k
```

```
real aamax,dum,sum,vv(NMAX)
```

```
d=1.
```

```
do i=1,n
```

```
aamax=0.
```

```
do j=1,n
```

```
if(abs(a(i,j)).gt.aamax) aamax=abs(a(i,j))
```

```
end do
```

```
if(aamax.eq.0) pause 'singular matrix in ludcmp'
```

```
vv(i)=1./aamax
```

```
end do
```

```
do j=1,n
```

```
do i=1,j-1
```

```
sum=a(i,j)
```

```
do k=1,i-1
```

```
sum=sum-a(i,k)*a(k,j)
```

```
end do
```

```
a(i,j)=sum
```

```
end do
```

```
aamax=0.
```

```
do i=j,n
```

```
sum=a(i,j)
```

```
do k=1,j-1
```

```
sum=sum-a(i,k)*a(k,j)
```

```
enddo
```

```
a(i,j)=sum
```

```
dum=vv(i)*abs(sum)
```

```
if(dum.ge.aamax)then
```

```
imax=i
```

```
aamax=dum
```

```
endif
```

```
enddo
```

```
if(j.ne.imax)then
```

```
do k=1,n
```

```
dum=a(imax,k)
```

```
a(imax,k)=a(j,k)
```

```
a(j,k)=dum
enddo
d=-d
vv(imax)=vv(j)
endif
indx(j)=imax
if(a(j,i).eq.0.)a(j,j)=tiny
if(j.ne.n)then
dum=1./a(j,j)
do i=j+1,n
a(i,j)=a(i,j)*dum
enddo
endif
enddo
return
end
```

```
subroutine lubksb(a,n,np,indx,b)
```

```
integer n,np,indx(n)
real a(np,np),b(n)
integer i,ii,j,ll
real sum
ii=0
do i=1,n
ll=indx(i)
sum=b(ll)
b(ll)=b(i)
if(ii.ne.0)then
do j=ii,i-1
sum=sum-a(i,j)*b(j)
enddo
else if(sum.ne.0.)then
ii=i
endif
b(i)=sum
enddo
do i=n,1,-1
sum=b(i)
do j=i+1,n
sum=sum-a(i,j)*b(j)
enddo
b(i)=sum/a(i,i)
enddo
return
end
```

APÊNDICE III – COMANDOS DE AJUSTES DE PLOTAGEM (GNU PLOT)

COMANDO PARA GRAFICAR NO GNU PLOT _ITAIPU_E6_SUMMER_11_01_2010

```
!*****  
! COMANDOS PARA GRAFICAR O CAMPO DE TEMPERATURA  
!*****  
set pm3d map  
set autoscale fix  
set palette defined (20 'white', 22 'blue', 28 'green', 30 'yellow', 32 'orange', 33 'red')  
unset key  
plot 'SAIDA1_THERMAL_GRAPHIC_MATRIX.txt' matrix using ($1+2):($2+2):3 with image  
  
!*****  
! COMANDOS PARA GRAFICAR O CAMPO DE DESLOCAMENTOS TÉRMICOS NA DIREÇÃO X  
!*****  
set pm3d map  
set autoscale fix  
set palette defined (-0.0005'white', 0 'blue', 0.3'green', 0.6 'yellow', 1.4 'orange', 1.8'red')  
unset key  
plot 'SAIDA2_X_TS_DISP_GRAPHIC_MATRIX.txt' matrix using ($1+2):($2+2):3 with image  
  
!*****  
! COMANDOS PARA GRAFICAR O CAMPO DE DESLOCAMENTOS TÉRMICOS NA DIREÇÃO Y  
!*****  
set pm3d map  
set autoscale fix  
set palette defined (-0.01'white', -0.001'blue', 0.3'green', 0.5'yellow', 0.9'orange', 1.2'red' )  
unset key  
plot 'SAIDA3_Y_TS_DISP_GRAPHIC_MATRIX.txt' matrix using ($1+2):($2+2):3 with image
```


COMANDO PARA GRAFICAR NO GNUPLOT__ITAIPU_E6_WINTER

```
!*****
```

```
! COMANDOS PARA GRAFICAR O CAMPO DE TEMPERATURA
```

```
!*****
```

```
set pm3d map
```

```
set autoscale fix
```

```
set palette defined (14 'white', 16 'blue', 17 'green', 18 'yellow', 19 'orange', 20.18 'red')
```

```
unset key
```

```
plot 'SAIDA1_THERMAL_GRAPHIC_MATRIX.txt' matrix using ($1+2):($2+2):3 with image
```

```
!*****
```

```
! COMANDOS PARA GRAFICAR O CAMPO DE DESLOCAMENTOS TÉRMICOS NA DIREÇÃO X
```

```
!*****
```

```
set pm3d map
```

```
set autoscale fix
```

```
set palette defined (-20.0 'red', -15.0 'orange', -10.0 'yellow', -5.0 'green', 0 'blue', 0.01 'white')
```

```
unset key
```

```
plot 'SAIDA2_X_TS_DISP_GRAPHIC_MATRIX.txt' matrix using ($1+2):($2+2):3 with image
```

```
!*****
```

```
! COMANDOS PARA GRAFICAR O CAMPO DE DESLOCAMENTOS TÉRMICOS NA DIREÇÃO Y
```

```
!*****
```

```
set pm3d map
```

```
set autoscale fix
```

```
set palette defined (-10 'red', -5 'orange', -4 'yellow', -3 'green', 1.4 'blue', 2.0 'white')
```

```
unset key
```

```
plot 'SAIDA3_Y_TS_DISP_GRAPHIC_MATRIX.txt' matrix using ($1+2):($2+2):3 with image
```

ANEXO I – SUBROTINAS FORTRAN (INVERTER MATRIZES)

```
SUBROUTINE ludcmp(a,n,np,indx,d)
  integer n,np,indx(n),nmax
  real d,a(np,np),TINY
  parameter (nmax=640,Tiny=1.0e-20)
  integer i,imax,j,k
  real aamax,dum,sum,vv(NMAX)
  d=1.
  do i=1,n
    aamax=0.
    do j=1,n
      if(abs(a(i,j)).gt.aamax) aamax=abs(a(i,j))
    end do
    if(aamax.eq.0) pause 'singular matrix in ludcmp'
    vv(i)=1./aamax
  end do

  do j=1,n
    do i=1,j-1
      sum=a(i,j)
      do k=1,i-1
        sum=sum-a(i,k)*a(k,j)
      end do
      a(i,j)=sum
    end do
    aamax=0.
    do i=j,n
      sum=a(i,j)
      do k=1,j-1
        sum=sum-a(i,k)*a(k,j)
      enddo
      a(i,j)=sum
      dum=vv(i)*abs(sum)
      if(dum.ge.aamax)then
        imax=i
        aamax=dum
      endif
    enddo
    if(j.ne.imax)then
      do k=1,n
        dum=a(imax,k)
        a(imax,k)=a(j,k)
        a(j,k)=dum
      enddo
      d=-d
      vv(imax)=vv(j)
    enddo
  enddo
```

```
endif
indx(j)=imax
if(a(j,j).eq.0.)a(j,j)=tiny
if(j.ne.n)then
dum=1./a(j,j)
do i=j+1,n
a(i,j)=a(i,j)*dum
enddo
endif
enddo
return
end
```

```
subroutine lubksb(a,n,np,indx,b)
```

```
integer n,np,indx(n)
real a(np,np),b(n)
integer i,ii,j,ll
real sum
ii=0
do i=1,n
ll=indx(i)
sum=b(ll)
b(ll)=b(i)
if(ii.ne.0)then
do j=ii,i-1
sum=sum-a(i,j)*b(j)
enddo
else if(sum.ne.0)then
ii=i
endif
b(i)=sum
enddo
do i=n,1,-1
sum=b(i)
do j=i+1,n
sum=sum-a(i,j)*b(j)
enddo
b(i)=sum/a(i,i)
enddo
return
end
```