



UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA MECÂNICA

FELIPE AUGUSTO DE OLIVEIRA MOTTA

Software intuitivo para o ensino de transferência de calor

UBERLÂNDIA

2021

Software intuitivo para o ensino de transferência de calor

Trabalho de Conclusão de Curso apresentado ao Programa de Graduação em Engenharia Mecânica da Faculdade de Engenharia Mecânica da Universidade Federal de Uberlândia como requisito parcial para obtenção do título de Bacharel em Engenharia Mecânica.

Orientadora: Priscila Ferreira Barbosa de Sousa

UBERLÂNDIA

2021

RESUMO

O cálculo de distribuições de temperatura geralmente é feito utilizando métodos numéricos computacionais, devido à dificuldade de se trabalhar com métodos analíticos. Por esse motivo, o ensino de métodos de solução numérica para distribuições de temperaturas é de grande importância nas disciplinas de transferência de calor na graduação.

Neste trabalho foi desenvolvida uma interface de usuário didática para cálculo numérico da distribuição de temperaturas em geometrias de placa plana. O software calcula soluções em regime permanente e transiente, aplicando o método das diferenças finitas, nas formas explícita e implícita. Nele o usuário pode interagir ativamente com todas as variáveis do problema de forma fácil e intuitiva. Dessa forma, o objetivo é fornecer uma ferramenta capaz de auxiliar no ensino de transferência de calor, por meio da facilitação do contato inicial com métodos de solução numérica e da representação visual dos efeitos de cada parâmetro na distribuição de temperaturas.

Palavras-chave: transferência de calor, distribuição de temperaturas, solução numérica, ensino de transferência de calor.

SUMÁRIO

1 INTRODUÇÃO	6
2 MÉTODOS	7
2.1 CARACTERÍSTICAS DO PROBLEMA.....	8
2.2 SOLUÇÃO NUMÉRICA	10
2.3 CRIAÇÃO DA INTERFACE COM O GUIDE	16
3 RESULTADOS	20
4 VALIDAÇÃO DOS RESULTADOS	22
4.1 TESTE 1.....	23
4.2 TESTE 2.....	26
4.3 TESTE 3.....	28
5 CONCLUSÕES	31
5.1 SUGESTÕES DE TRABALHOS FUTUROS	31
BIBLIOGRAFIA.....	32

LISTA DE ILUSTRAÇÕES

Figura 1 - Representação de malha.....	10
Figura 2 - Representação de um nó no interior da malha.....	10
Figura 3 – Volume de controle em ponto na extremidade superior direita.....	12
Figura 4 - Tela principal do programa, com solução de problema exemplo.	16
Figura 5 - Espaço de trabalho da extensão do Matlab, GUIDE.....	17
Figura 6 - Tela inicial do programa e código escrito para seu funcionamento.	17
Figura 7 - Tela principal do programa.....	19
Figura 8 - Mensagem de erro referente a critério de estabilidade	21
Figura 9 - Tela para análise de temperatura de ponto em regime permanente.	22
Figura 10 - Tela para análise de temperatura de ponto em regime transiente.	22
Figura 11 - Resultados do CTPP para teste de transferência de calor em aleta, caso A. .	24
Figura 12 - Gráficos de comparação dos resultados do CTPP com soluções analíticas, em transferência de calor em aletas.	25
Figura 13 - Esquema do exemplo 5.4 de Incropera.	26
Figura 14 - Tela principal do CTPP em solução do teste 2.	27
Figura 15 - Análise de temperatura em ponto em solução do teste 2.	27
Figura 16 - Resultados de distribuição de temperatura em teste 3, solução analítica.....	28
Figura 17 - Resultados da distribuição de temperaturas em teste 3, CTPP.	29
Figura 18 – Comparação entre software analítico e CTPP.	30

1 INTRODUÇÃO

O foco principal no estudo de transferência de calor é a determinação de taxas de transferência de energia térmica, o que é diferente de uma análise termodinâmica, pois esta trata das quantidades totais de energia transferida quando há mudanças nas condições de um sistema, sem se atentar ao tempo ou a forma como o calor é perdido ou ganho. Outra aplicação fundamental da transferência de calor é na obtenção de distribuições de temperatura. Frequentemente há interesse em se determinar a distribuição de temperatura para um corpo ou mesmo a região de concentração de maiores temperatura. Pode-se citar o caso de processos de fabricação onde geração excessiva de calor pode causar danos à peça, à ferramenta ou mesmo ao processo (BONNAND, 2017).

Soluções analíticas fornecem equações que descrevem a distribuição de temperatura. Estas equações são obtidas por meio da solução de equações diferenciais, quando são conhecidas as condições iniciais e de contorno. Soluções analíticas são exatas, porém são limitadas a problemas com condições de contorno homogêneas e geometrias básicas que possam ser representadas perfeitamente num sistema de coordenadas, como planos, esferas e cilindros (ÇENGEL, 2012). Quando isso não ocorre, o problema pode ser simplificado para se adequar a um método analítico. Nesse contexto, soluções numéricas são de suma importância na análise de problemas de transferência de calor mais complexos nos quais a solução analítica é de difícil obtenção.

Atualmente, encontra-se no mercado vários softwares para solução numérica de problemas térmicos. Estes softwares possibilitam a simulação de problemas térmicos de alta ou baixa complexidade sem exigir que o usuário tenha domínio das técnicas de solução numéricas. Contudo o usuário ainda precisa interpretar o problema e seus resultados. Assim, o aprendizado das soluções analíticas e numéricas se mantém indispensável. No ambiente acadêmico, especificamente na graduação, uma ferramenta de simulação simples e acessível, capaz de resolver problemas fundamentais deve contribuir para a melhoria do aprendizado em transferência de calor, ao possibilitar a interação com as variáveis do problema, visualização dos efeitos térmicos e comparação de resultados.

2 MÉTODOS

O presente trabalho consiste de uma interface de usuário didática, para uso em sala de aula, capaz de resolver problemas de transferência de calor em placa plana. Para o desenvolvimento da interface foram usados como base dois códigos criados por alunos da Faculdade de Engenharia Mecânica na disciplina Transferência de Calor, que aplicam o método de diferenças finitas, um na forma implícita e outro na forma explícita. Os códigos foram modificados e adaptados para integração com a interface de usuário desenvolvida. É possível obter a distribuição de temperaturas tanto em regime permanente quanto transiente.

Além da distribuição de temperaturas na placa, a validação numérica dos resultados pode ser verificada por meio de um balanço de energia, que é calculado após cada solução. Além disso, neste trabalho foram feitos testes de validação, por meio da comparação dos resultados das soluções numéricas com soluções analíticas.

A interface foi criada no Matlab, versão R2015a, com utilização da extensão GUIDE. A escolha da GUIDE se deve à facilidade de estruturação da interface e também pela possibilidade de criação de um aplicativo para uso independente, na forma de pacote de instalação. Para utilizar o software, o usuário deve fazer a instalação do pacote em seu computador, porém não é necessário possuir o Matlab, o que facilita a distribuição.

É importante lembrar que a extensão GUIDE está sendo substituída por uma mais moderna, chamada App Designer, e será removida em uma versão futura do Matlab. Aplicativos criados no GUIDE continuarão funcionando, mas não poderão ser editados futuramente. A migração dos aplicativos do GUIDE para App Designer pode ser feita de forma semiautomática a partir da versão R2018a do Matlab.

2.1 CARACTERÍSTICAS DO PROBLEMA

Os seguintes mecanismos de transferência de calor estão envolvidos nos problemas que o software é capaz de resolver:

CONDUÇÃO

É o modo de transferência de calor que ocorre devido a um conjunto de interações entre partículas como colisões, difusão, vibração e transferência de elétrons, gerando uma transferência de energia das partículas mais energéticas para as menos energéticas. A condução é responsável pela transferência de calor no interior da placa plana do software. A taxa de transferência de calor por condução unidimensional na direção x é determinada pela Lei de Fourier:

$$q = -kA \frac{dT}{dx} \quad (1)$$

Onde,

k Condutividade térmica em $W/(m.K)$

A Área onde se localiza o fluxo de calor, em m^2

dT/dx Gradiente de temperatura na direção x

CONVECÇÃO

Transferência de calor que ocorre na interação entre uma superfície e um fluido, envolvendo os fenômenos de difusão e advecção. Este modo de transferência de calor poderá ocorrer nas fronteiras como condição de contorno por meio da Lei do Resfriamento de Newton:

$$q = hA(T_s - T_\infty) \quad (2)$$

Onde,

h Coeficiente de transferência de calor por convecção, em $W/(m^2K)$

T_s Temperatura na superfície do sólido, em K

T_∞ Temperatura do fluido, em K

GERAÇÃO DE CALOR E FLUXO DE CALOR

O software possibilita a inclusão da geração de energia térmica nos problemas. A ocorrência de geração de calor numa placa pode ser exemplificada em processos nos quais ocorrem conversão de alguma forma de energia em energia térmica, como em situações de aquecimento por efeito Joule e absorção de radiação (INCROPERA, 2008). No software, deve ser considerada uma taxa de geração de calor uniforme no volume da placa, em W/m^3 .

Também podem ser incluídos fluxos de calor nas paredes da placa. O fluxo deve ser permanente e uniforme em cada parede, colocado na forma W/m^2 (O software não permite a entrada de fluxo com variação no tempo ou não uniforme). Uma situação capaz de gerar esse efeito seria, por exemplo, a fixação de um fino aquecedor elétrico sobre toda a superfície da parede.

Ao inserir os dados do problema, o usuário define as condições de contorno e inicial do problema a ser abordado. Em cada parede as possíveis condições de contorno são as seguintes (INCROPERA, 2008), considerando aqui uma parede em $x=0$:

- Superfície com temperatura constante: $T(0, t) = T_s$ (3)

- Fluxo térmico diferente de zero: $-k \frac{\partial T}{\partial x} \Big|_{x=0} = q_s''$ (4)

- Superfície isolada ou adiabática: $\frac{\partial T}{\partial x} \Big|_{x=0} = 0$ (5)

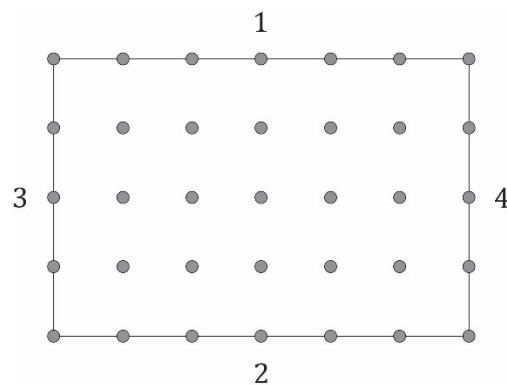
- Superfície sob convecção: $-k \frac{\partial T}{\partial x} \Big|_{x=0} = h[T_\infty - T(0, t)]$ (6)

- Superfície sob convecção e fluxo térmico: $-k \frac{\partial T}{\partial x} \Big|_{x=0} = h[T_\infty - T(0, t)] + q_s''$ (7)

2.2 SOLUÇÃO NUMÉRICA

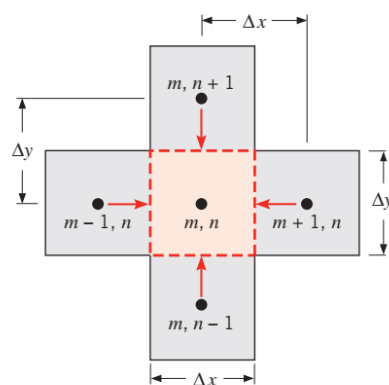
O problema inserido pelo usuário é resolvido utilizando o método do balanço de energia usando diferenças finitas. Para isso, a placa plana é discretizada em um conjunto finito de pontos ou nós, formando uma malha bidimensional. A figura 1 mostra uma representação da placa discretizada e a numeração utilizada para referenciar os lados da placa. As temperaturas são calculadas em cada nó. Um ponto do interior da malha é representado na figura 2.

Figura 1 - Representação de malha



Fonte: Próprio autor.

Figura 2 - Representação de um nó no interior da malha.



Fonte: Incropera (2008, p.137).

A Figura 2 mostra o ponto (m,n) e seus pontos adjacentes. A posição vertical de um ponto é representada pelo índice n e a horizontal por m . Cada ponto ou nó representa um elemento de dimensões Δx e Δy . Na solução numérica utilizada no software, foi considerado $\Delta x = \Delta y$. Esta dimensão é função do número de nós na direção horizontal,

valor escolhido pelo usuário, que define o refinamento da malha. O usuário também escolhe o tempo de simulação e o valor de Δt , o intervalo de tempo. As temperaturas são calculadas em cada ponto, em instantes de tempo da forma Δt , $2\Delta t$, $3\Delta t$, sucessivamente. Assim, o problema é discretizado no tempo e no espaço.

O próximo passo para obtenção da temperatura no ponto (m,n) é formular o balanço de energia que assume a seguinte forma:

$$\dot{E}_{entra} + \dot{E}_{gerada} = \dot{E}_{acumulada} \quad (8)$$

Na equação (8) o termo \dot{E}_{entra} inclui quatro taxas, referentes à troca de calor por condução do ponto (m,n) com os pontos acima, abaixo, à esquerda e à direita. Como seus sentidos são desconhecidos, toda transferência de calor é considerada entrando no ponto (m,n), convenção do método.

A taxa direcionada do ponto (m-1,n) para o ponto (m,n), pela lei de Fourier é dada por:

$$q_{(m-1,n) \rightarrow (m,n)} = k(\Delta y \cdot 1) \frac{T_{m-1,n} - T_{m,n}}{\Delta x} \quad (9)$$

Onde no problema bidimensional, Δz é considerado unitário. As equações das outras taxas assumem formas semelhantes.

O termo \dot{E}_{gerada} é a geração de calor no interior do volume de controle, definido por:

$$\dot{E}_g = q_g \cdot \Delta x \Delta y \Delta z \quad (10)$$

Do lado direito da equação (8), $\dot{E}_{acumulada}$ representa o termo de acumulação de energia no tempo:

$$\dot{E}_a = \rho C_p \Delta x \Delta y \frac{T_{m,n}^{t+\Delta t} - T_{m,n}^t}{\Delta t} \quad (11)$$

Onde foi feita uma discretização no tempo da derivada $\partial T / \partial t$.

No método explícito, a temperatura é calculada num ponto no tempo (t+ Δt) em função das temperaturas dos pontos adjacentes no instante de tempo anterior, t. Dessa forma, as temperaturas de cada ponto podem ser calculadas diretamente, de forma sucessiva, a partir da temperatura inicial conhecida. O balanço de energia do ponto (m,n), que representa um nó no interior do sólido, no método explícito resulta em:

$$\begin{aligned}
& k\Delta y \frac{T_{m-1,n}^t - T_{m,n}^t}{\Delta x} + k\Delta x \frac{T_{m,n-1}^t - T_{m,n}^t}{\Delta y} + k\Delta y \frac{T_{m+1,n}^t - T_{m,n}^t}{\Delta x} \\
& + k\Delta x \frac{T_{m,n+1}^t - T_{m,n}^t}{\Delta y} + q_g \Delta x \Delta y = \rho C_p \Delta x \Delta y \frac{T_{m,n}^{t+\Delta t} - T_{m,n}^t}{\Delta t}
\end{aligned} \quad (12)$$

No caso onde $\Delta x = \Delta y$, a equação (12) pode ser manipulada para obter-se:

$$T_{m,n}^{t+\Delta t} = Fo(T_{m-1,n}^t + T_{m,n-1}^t + T_{m+1,n}^t + T_{m,n+1}^t) + (1 - 4Fo)T_{m,n}^t + \frac{q_g \Delta t}{\rho C_p} \quad (13)$$

$$\text{Onde } Fo = \frac{k\Delta t}{\rho C_p \Delta x^2}$$

No método implícito as temperaturas são calculadas em um ponto, no tempo $(t+\Delta t)$ em função das temperaturas dos pontos adjacentes, também em $(t+\Delta t)$. Na forma implícita, o balanço de energia para o ponto (m,n) resulta em:

$$\begin{aligned}
& k\Delta y \frac{T_{m-1,n}^{t+\Delta t} - T_{m,n}^{t+\Delta t}}{\Delta x} + k\Delta x \frac{T_{m,n-1}^{t+\Delta t} - T_{m,n}^{t+\Delta t}}{\Delta y} + k\Delta y \frac{T_{m+1,n}^{t+\Delta t} - T_{m,n}^{t+\Delta t}}{\Delta x} \\
& + k\Delta x \frac{T_{m,n+1}^{t+\Delta t} - T_{m,n}^{t+\Delta t}}{\Delta y} + q_g \Delta x \Delta y = \rho C_p \Delta x \Delta y \frac{T_{m,n}^{t+\Delta t} - T_{m,n}^t}{\Delta t}
\end{aligned} \quad (14)$$

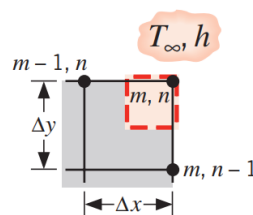
Fazendo $\Delta x = \Delta y$, a equação (14) é manipulada para obter-se:

$$T_{m,n}^t + \frac{q_g \Delta t}{\rho C_p} = -Fo(T_{m-1,n}^{t+\Delta t} + T_{m,n-1}^{t+\Delta t} + T_{m+1,n}^{t+\Delta t} + T_{m,n+1}^{t+\Delta t}) + (4Fo + 1)T_{m,n}^{t+\Delta t} \quad (15)$$

Portanto, tem-se um sistema linear e as equações correspondentes a cada ponto da malha devem ser resolvidas simultaneamente (BONNAND, 2017).

O balanço de energia deve ser feito de forma semelhante em cada ponto para obter as equações correspondentes. Como exemplo, a seguir será mostrada a resolução do balanço de energia para o ponto da extremidade superior direita, representado na figura 3.

Figura 3 – Volume de controle em ponto na extremidade superior direita.



Fonte: Incropera (2008, p.190).

O primeiro termo do balanço de energia, referente a fluxos de energia entrando no volume de controle, neste caso inclui as taxas de transferência de calor por condução por pontos adjacentes da placa e também as taxas de transferência de calor devido convecção e por fluxo uniforme de calor em ambos lados da extremidade.

$$\dot{E}_{entra} + \dot{E}_{gerada} = \dot{E}_{acumulada} \quad (16)$$

$$\begin{aligned} \dot{E}_{entra} = k \left(\frac{\Delta x}{2} \right) \frac{T_{m,n-1} - T_{m,n}}{\Delta y} + k \left(\frac{\Delta y}{2} \right) \frac{T_{m-1,n} - T_{m,n}}{\Delta x} + h \left(\frac{\Delta x}{2} \right) (T_{\infty 1} - T_{m,n}) \\ + h \left(\frac{\Delta y}{2} \right) (T_{\infty 4} - T_{m,n}) + q_1 \frac{\Delta x}{2} + q_4 \frac{\Delta y}{2} \end{aligned} \quad (17)$$

$$\dot{E}_g = \frac{q_g \cdot \Delta x \Delta y}{4} \quad (18)$$

$$\dot{E}_a = \frac{\rho C_p \Delta x \Delta y}{4} \left(\frac{T_{m,n}^{t+\Delta t} - T_{m,n}^t}{\Delta t} \right) \quad (19)$$

Onde $T_{\infty 1}$ e $T_{\infty 4}$ são as temperaturas do fluido externo nos lados superior e direito, respectivamente.

E q_1 e q_4 são os fluxos constantes de calor nos lados superior e direito, respectivamente.

Os termos, após substituídos na equação (16), podem ser manipulados para resultar na equação (20), no método explícito:

$$\begin{aligned} T_{m,n}^{t+\Delta t} = 2Fo \left(T_{m,n-1}^t + T_{m-1,n}^t + Bi_1 T_{\infty 1} + Bi_4 T_{\infty 4} + (q_1 + q_4) \frac{\Delta x}{k} \right) \\ + (1 - 4Fo - 2Bi_1 Fo - 2Bi_4 Fo) T_{m,n}^t + \frac{q_g \Delta t}{\rho C_p} \end{aligned} \quad (20)$$

Todas as equações utilizadas no programa são mostradas a seguir, no método explícito. Nelas foi considerada a possibilidade de diferentes condições de convecção e fluxo de calor em cada lado da placa.

Interior:

$$T_{m,n}^{t+\Delta t} = Fo (T_{m-1,n}^t + T_{m,n-1}^t + T_{m+1,n}^t + T_{m,n+1}^t) + (1 - 4Fo) T_{m,n}^t + \frac{q_g \Delta t}{\rho C_p} \quad (21)$$

Parede superior:

$$T_{m,n}^{t+\Delta t} = Fo \left(2T_{m,n-1}^t + T_{m-1,n}^t + T_{m+1,n}^t + 2Bi_1T_{\infty 1} + \frac{2q_1\Delta x}{k} \right) + (1 - 4Fo - 2Bi_1Fo)T_{m,n}^t + \frac{q_g\Delta t}{\rho C_p} \quad (22)$$

Parede inferior:

$$T_{m,n}^{t+\Delta t} = Fo \left(2T_{m,n+1}^t + T_{m-1,n}^t + T_{m+1,n}^t + 2Bi_2T_{\infty 2} + \frac{2q_2\Delta x}{k} \right) + (1 - 4Fo - 2Bi_2Fo)T_{m,n}^t + \frac{q_g\Delta t}{\rho C_p} \quad (23)$$

Parede esquerda:

$$T_{m,n}^{t+\Delta t} = Fo \left(2T_{m+1,n}^t + T_{m,n-1}^t + T_{m,n+1}^t + 2Bi_3T_{\infty 3} + \frac{2q_3\Delta x}{k} \right) + (1 - 4Fo - 2Bi_3Fo)T_{m,n}^t + \frac{q_g\Delta t}{\rho C_p} \quad (24)$$

Parede direita:

$$T_{m,n}^{t+\Delta t} = Fo \left(2T_{m-1,n}^t + T_{m,n-1}^t + T_{m,n+1}^t + 2Bi_4T_{\infty 4} + \frac{2q_4\Delta x}{k} \right) + (1 - 4Fo - 2Bi_4Fo)T_{m,n}^t + \frac{q_g\Delta t}{\rho C_p} \quad (25)$$

Quina superior esquerda:

$$T_{m,n}^{t+\Delta t} = 2Fo \left(T_{m,n-1}^t + T_{m+1,n}^t + Bi_1T_{\infty 1} + Bi_3T_{\infty 3} + (q_1 + q_3) \frac{\Delta x}{k} \right) + (1 - 4Fo - 2Bi_1Fo - 2Bi_3Fo)T_{m,n}^t + \frac{q_g\Delta t}{\rho C_p} \quad (26)$$

Quina superior direita:

$$T_{m,n}^{t+\Delta t} = 2Fo \left(T_{m,n-1}^t + T_{m-1,n}^t + Bi_1T_{\infty 1} + Bi_4T_{\infty 4} + (q_1 + q_4) \frac{\Delta x}{k} \right) + (1 - 4Fo - 2Bi_1Fo - 2Bi_4Fo)T_{m,n}^t + \frac{q_g\Delta t}{\rho C_p} \quad (27)$$

Quina inferior esquerda:

$$T_{m,n}^{t+\Delta t} = 2Fo \left(T_{m,n+1}^t + T_{m+1,n}^t + Bi_3T_{\infty 3} + Bi_2T_{\infty 2} + (q_2 + q_3) \frac{\Delta x}{k} \right) + (1 - 4Fo - 2Bi_2Fo - 2Bi_3Fo)T_{m,n}^t + \frac{q_g\Delta t}{\rho C_p} \quad (28)$$

Quina inferior direita:

$$T_{m,n}^{t+\Delta t} = 2Fo \left(T_{m,n+1}^t + T_{m-1,n}^t + Bi_2 T_{\infty_2} + Bi_4 T_{\infty_4} + (q_2 + q_4) \frac{\Delta x}{k} \right) + (1 - 4Fo - 2Bi_2 Fo - 2Bi_4 Fo) T_{m,n}^t + \frac{q_g \Delta t}{\rho C_p} \quad (29)$$

É importante destacar uma vantagem do método implícito em relação ao explícito: o método implícito é incondicionalmente estável, ou seja, ele converge a um resultado independentemente dos valores da malha (Δx e Δy) e do passo no tempo (Δt). O método explícito, em contrapartida, apresenta restrições quanto a esses valores e para garantir a convergência o critério de estabilidade deve ser garantido. De acordo com Incropera (2008, p.189) “ o critério é determinado pela exigência de que o coeficiente associado ao nó de interesse no instante anterior seja maior ou igual a zero”.

Na equação (21) para um ponto interno, portanto, o critério de estabilidade é obtido da seguinte forma:

$$(1 - 4Fo) \geq 0 \quad (30)$$

Então:

$$Fo \leq 1/4 \quad (31)$$

Na placa plana também devem ser considerados os critérios de estabilidade dos pontos de cada parede e quina, obtendo assim 9 condições para estabilidade. Por exemplo, para um ponto na parede, o critério de estabilidade exige que:

$$(1 - 4Fo - 2BiFo) \geq 0 \quad (32)$$

Então:

$$Fo(2 + Bi) \leq 1/2 \quad (33)$$

Onde $Fo = \frac{k\Delta t}{\rho C_p \Delta x^2}$ e $Bi = \frac{h\Delta x}{k}$

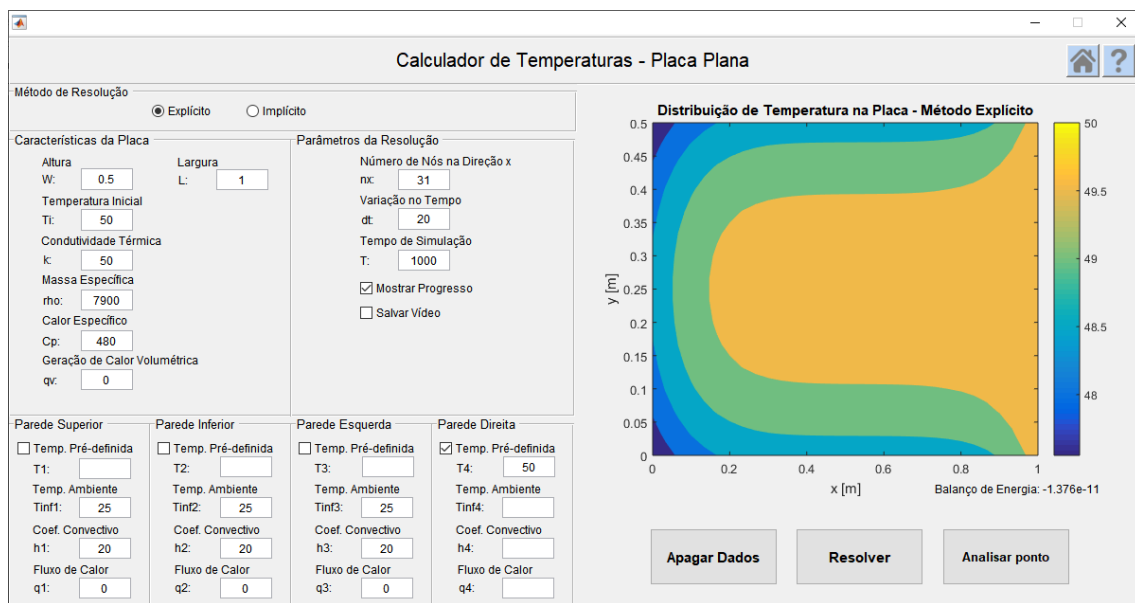
Como Fo e Bi são funções de Δx e Δt , estes valores devem ser ajustados para garantir a estabilidade da solução. De forma geral, para um dado valor fixo de Δx , o valor de Δt deve ficar abaixo de um limite estabelecido pelo critério mais restritivo.

Após calculada a solução por um dos métodos, o programa apresenta a distribuição de temperatura. A validade da solução pode ser verificada por meio do resultado de um

balanço de energia em uma superfície de controle envolvendo todas as regiões nodais cujas temperaturas foram determinadas.

2.3 CRIAÇÃO DA INTERFACE COM O GUIDE

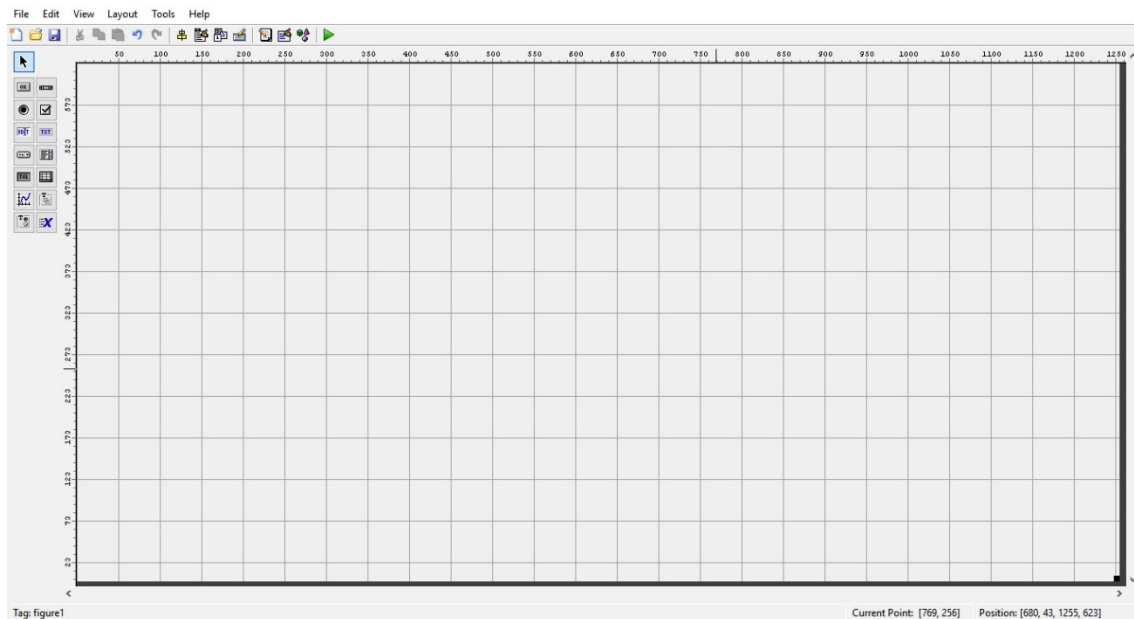
Figura 4 - Tela principal do programa, com solução de problema exemplo.



Fonte: Próprio autor.

A figura 4 mostra a tela principal do software, com dados e resultados de problema exemplo, o qual é pré-inserido ao início do programa. O software foi criado utilizando o *GUIDE*, um ambiente de criação de interfaces de usuário vinculado ao Matlab. O desenvolvimento de um aplicativo com essa ferramenta é bastante prático pois a criação da interface é feita em um espaço de trabalho (figura 5) onde o usuário pode construir toda a parte visual sem necessitar trabalhar com códigos inicialmente. Nesse espaço de trabalho do *GUIDE* cria-se o layout da interface, pela adição e posicionamento de componentes como caixas de texto, botões, caixas de seleção, barras de rolagem, etc. Essa configuração do layout resume-se a escolher os componentes disponíveis num menu lateral, posicioná-los no local desejado, redimensioná-los e escolher propriedades como cor, ativação, visibilidade, etc.

Figura 5 - Espaço de trabalho da extensão do Matlab, GUIDE.

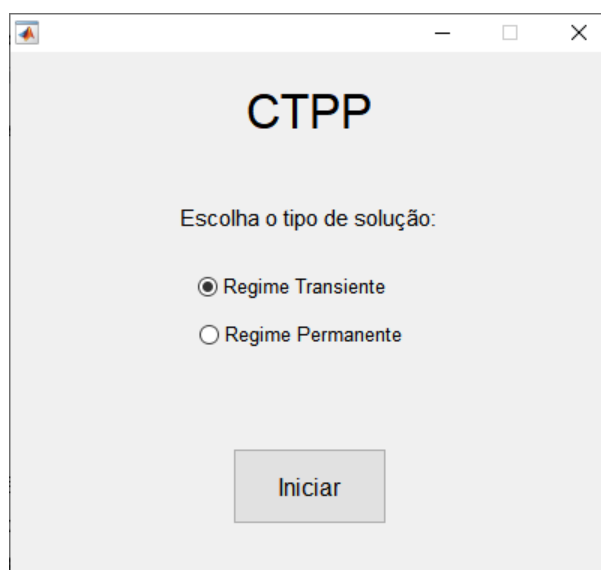


Fonte: Matlab.

Enquanto o layout da interface é criado nesse espaço de trabalho visual, a lógica do programa é configurada em uma função do Matlab vinculada à interface. Cada componente da interface possui um *tag* para identificação. Através do *tag*, pode-se referenciar um componente da interface no código e alterar suas propriedades e comportamento.

Para exemplificar, a figura 6 mostra a tela inicial do programa e abaixo todo o código que foi escrito para programar seu funcionamento.

Figura 6 - Tela inicial do programa e código escrito para seu funcionamento.



```

76  □ function bot_inic_Callback(hObject, eventdata, handles)
77
78  -   r_selec = get(handles.selec_reg, 'SelectedObject');
79  -   reg_selec = get(r_selec, 'String');
80  -   setappdata(0, 'reg_seleccionado', reg_selec);
81
82  -   teste1_GUI
83  -   closereq();

```

Fonte: Próprio autor.

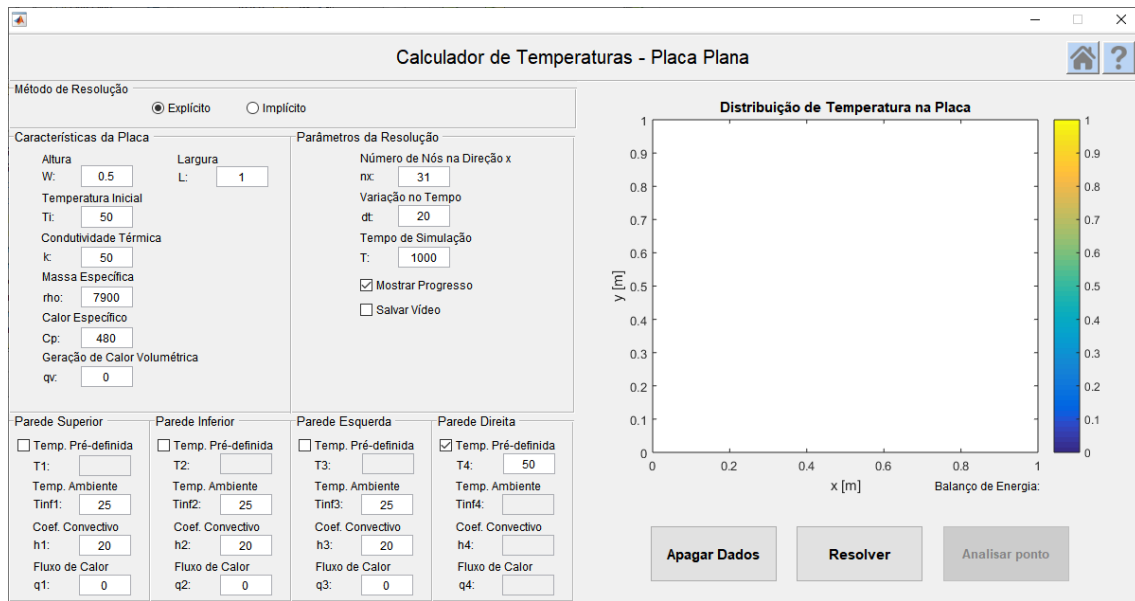
A linha 76 é pré escrita pelo Matlab quando o botão “Iniciar” é colocado na interface. É a chamada de uma função de “Callback”, na qual qualquer comando escrito abaixo será executado quando o botão de *tag* “bot_inic” for pressionado. Portanto, o que ocorre ao se pressionar o botão iniciar é o seguinte:

- A função “*get(handles.selec_reg, 'SelectedObject')*” na linha 78 lê qual opção de regime está selecionada atualmente, através da obtenção da propriedade “*SelectedObject*” do componente de *tag* “*selec_reg*”, correspondente ao botão de opções. A opção selecionada é atribuída à variável “*r_selec*”.
- Na linha 79, a função “*get(r_selec, 'String')*” obtém o texto correspondente à opção selecionada (Regime Transiente ou Regime Permanente), guardado na propriedade “*String*” da variável “*r_selec*” e coloca-o em outra variável “*reg_selec*”. Na linha 80, esse texto é guardado nas propriedades da interface para uso posterior.
- Na linha 82 está a chamada da função correspondente à tela principal do programa, “*teste1_GUI*”. Ao executar essa função, a tela principal é aberta e assim que isso ocorre, a função “*closereq()*” na linha 83 fecha a tela inicial.

A figura 7 mostra a tela principal do programa, em regime transiente. Nesta tela o usuário deve preencher todas as informações do problema que se deseja resolver, sendo elas:

- Características da Placa: Altura, largura, temperatura inicial, condutividade, térmica, massa específica, calor específico, geração de calor volumétrica.
- Parâmetros da Resolução: Número de nós na direção x, variação no tempo, tempo de simulação.
- Condições nas paredes: Temperatura prescrita na parede, temperatura ambiente, coeficiente convectivo, fluxo de calor imposto.

Figura 7 - Tela principal do programa.



Fonte: Próprio autor.

A principal função da interface é ler e coletar os dados inseridos pelo usuário, executar uma verificação de validade dos valores e fornecê-los às funções responsáveis pelo cálculo da distribuição de temperatura. Isso é feito no código principalmente por meio das funções “*get()*”, que lê alguma propriedade de um componente da interface, “*set()*”, usada para alterar uma propriedade do componente e também de funções de controle comuns em programação, como “*if*” e “*while*”, além de várias funções próprias do Matlab.

A seguir é mostrado como exemplo um trecho do código responsável por validar o tamanho da malha e mostrar uma mensagem de erro caso a malha seja muito grande.

```

W= str2double(get(handles.in_w, 'string'));
nx= str2double(get(handles.in_nx, 'string'));
ny= round(W/dx+1)
tam_malha = nx*ny

if tam_malha > 2601
    verif = 0;
    msgbox({'\fontsize{10}Malha muito grande, reduza nx ou inverta W e
L.';sprintf('(nx = %g, ny = %g)',nx_verif,ny_verif)},'Erro','error',
CreateStruct)
    break
end

```

Abaixo apresenta-se um trecho do código que verifica se não houveram erros de validação, recupera o regime selecionada pelo usuário na tela inicial, altera o título do gráfico da distribuição de temperaturas e fornece todas variáveis necessárias à função que calcula a solução pelo método implícito:

```
if verif == 1
    reg_selecionado = getappdata(0,'reg_selecionado');
    if reg_selecionado == 'Regime Permanente'
        permanente = 1;
        set(handles.titulo,'string','Distribuição de Temperatura na
Placa - Regime Permanente')
        [res_bal,nt,x,y,Tt,ntempo,tfinal,regime,nx,ny] =
perm_imp_func(L,W,Ti,k,rho,cp,qv,dt,nx,tfinal,tp1,tp2,tp3,tp4,T1,T2,T3
,T4,T_inf1,T_inf2,T_inf3,T_inf4,h1,h2,h3,h4,q1,q2,q3,q4,mostrarprog );
```

3 RESULTADOS

Esta seção dedica-se a apresentar o software desenvolvido neste trabalho e também a orientar sobre seu uso. O nome dado ao software é “Calculador de Temperaturas – Placa Plana”, ou de forma abreviada, CTPP.

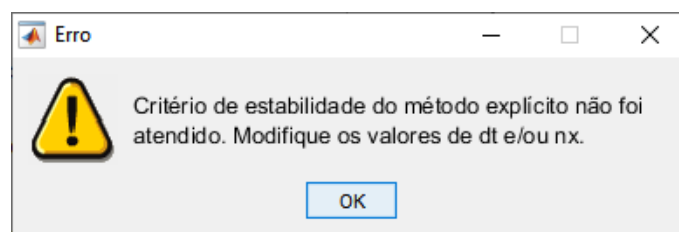
A distribuição do CTPP é feita por meio de um pacote de instalação gerado pelo compilador de aplicações do Matlab. A instalação é simples, semelhante à instalação dos softwares mais comuns do Windows, portanto, o usuário não deverá encontrar dificuldades. Durante o processo, será instalado também o “Matlab Runtime”, necessário para execução do CTPP e outros executáveis criados da mesma forma. Com esse método de instalação, não é necessário que o usuário possua o Matlab em seu dispositivo, o que facilita a distribuição do software.

Ao iniciar o software, o usuário deve escolher se deseja soluções em regime permanente ou transiente na tela inicial, figura 6.

Ao selecionar o regime, será aberta a interface principal do software. Na figura 7 apresenta-se a interface principal do regime transiente. Para regime permanente, a interface é semelhante, porém não existem os componentes para: seleção do tempo de simulação, diferencial de tempo, exibição e salvamento de vídeo. Além disso, em regime permanente, somente a solução pelo método implícito foi incluída.

Na interface principal, o usuário deve inserir todas as condições do problema de transferência de calor que se deseja resolver e os parâmetros para a simulação. Feito isso, pressiona-se o botão executar. Nessa etapa, o software verifica a validade dos dados inseridos. Por exemplo, caso tenha sido inserido um valor negativo para o comprimento da placa, o software exibirá uma mensagem de erro indicando o problema, para que o usuário o corrija e o software possa prosseguir com a execução. Também será verificado o critério de estabilidade (mensagem de erro mostrada na figura 8) em caso de solução pelo método explícito. O tamanho da malha depende do valor inserido pelo usuário em n_x (número de nós na direção x) e das dimensões da placa. Uma malha com 2601 nós é o tamanho máximo permitido. Essa restrição foi colocada para evitar travamentos.

Figura 8 - Mensagem de erro referente a critério de estabilidade

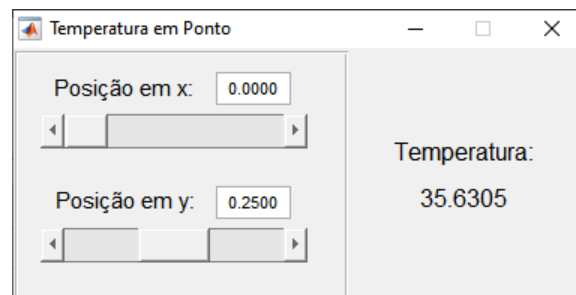


Fonte: Próprio autor.

Após verificados todos os dados, a simulação será executada. Em regime transiente uma animação mostrando a variação da distribuição de temperaturas com o tempo pode ser exibida, ou somente o resultado final, se o usuário assim preferir. Há também a opção de salvar essa animação no dispositivo. Abaixo do gráfico da distribuição de temperaturas observa-se o resultado do balanço de energia, que é a validação numérica da solução do problema.

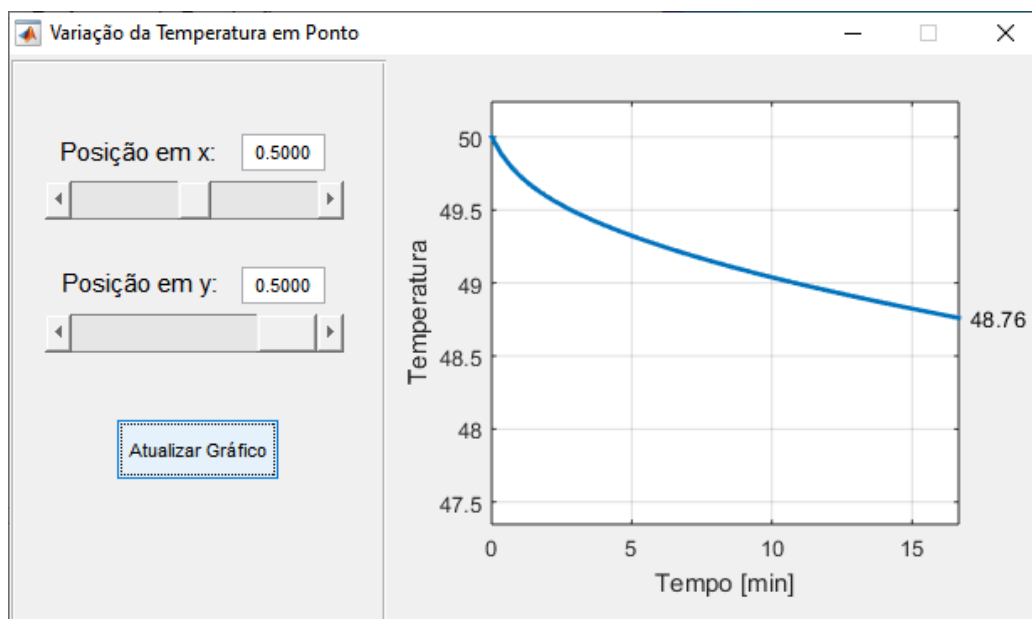
Com a simulação é concluída, o usuário pode clicar no botão "Analisar ponto". Esse botão abre uma nova janela, onde se pode selecionar um ponto específico da placa através das coordenadas x e y . Em regime permanente, a temperatura final do ponto é exibida. Em regime transiente é exibido um gráfico da temperatura do ponto em função do tempo. As duas telas são mostradas nas figuras 9 e 10.

Figura 9 - Tela para análise de temperatura de ponto em regime permanente.



Fonte: Próprio autor.

Figura 10 - Tela para análise de temperatura de ponto em regime transiente.



Fonte: Próprio autor.

4 VALIDAÇÃO DOS RESULTADOS

Nesta seção são mostrados os testes realizados para validação dos resultados do software. Três testes foram realizados:

1. Comparação com resultados analíticos em transferência de calor unidimensional em aleta plana de seção transversal uniforme, regime permanente.
2. Comparação com resultados analíticos aproximados de transferência de calor unidimensional transiente em parede plana.
3. Comparação com distribuição de temperatura de placa plana, em transferência de calor bidimensional transiente, obtida por software de solução analítica.

4.1 TESTE 1

Nesse teste foi feita a comparação da distribuição de temperaturas em diferentes casos de transferência de calor em aletas. Tais casos e suas respectivas equações analíticas para distribuição de temperaturas são os seguintes, conforme Incropera (2008, p.88):

Caso	Condições na Extremidade ($x=L$)	Distribuição de Temperaturas θ/θ_b
(A) Transferência de calor convectiva	$h\theta(L) = -kd\theta/dx _{x=L}$	$\frac{\cosh m(L-x) + (h/mk)\sinh m(L-x)}{\cosh mL + (h/mk)\sinh mL}$
(B) Adiabática	$d\theta/dx _{x=L} = 0$	$\frac{\cosh m(L-x)}{\cosh mL}$
(C) Temperatura especificada	$\theta(L) = \theta_L$	$\frac{(\theta_L/\theta_b)\sinh mx + \sinh m(L-x)}{\sinh mL}$
(D) Aleta infinita	$\theta(L) = 0$	e^{-mx}
$\theta \equiv T - T_\infty$ $m^2 \equiv hP/kA_{tr}$ $\theta_b = \theta(0) = T_b - T_\infty$		

As seguintes condições foram consideradas nas resoluções:

$$k = 237 \text{ W/(m. K)}$$

$$h = 100 \text{ W/(m}^2 \cdot \text{K)}$$

$$\rho = 2702 \text{ kg/m}^3$$

$$c = 903 \text{ J/(kg. K)}$$

$$W = 0,0025 \text{ m}$$

$$L = 0,05 \text{ m}$$

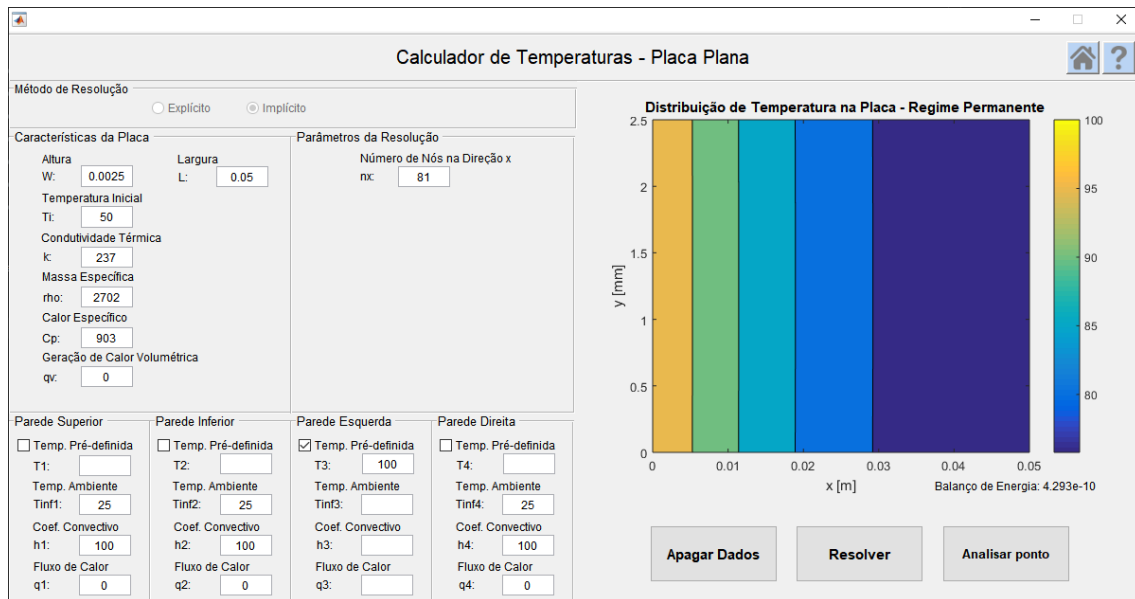
$$T_\infty = 25 \text{ }^\circ\text{C}$$

$$T_b = 100 \text{ }^\circ\text{C}$$

No caso C, com temperatura especificada na extremidade da aleta, a temperatura na extremidade T_L utilizada foi de 25. No caso D, aleta infinita, o comprimento L da aleta utilizado foi de 0,25m, um valor 100 vezes maior que a espessura da aleta.

Como exemplo, na figura 11 é mostrada a interface do CTPP na solução do caso A:

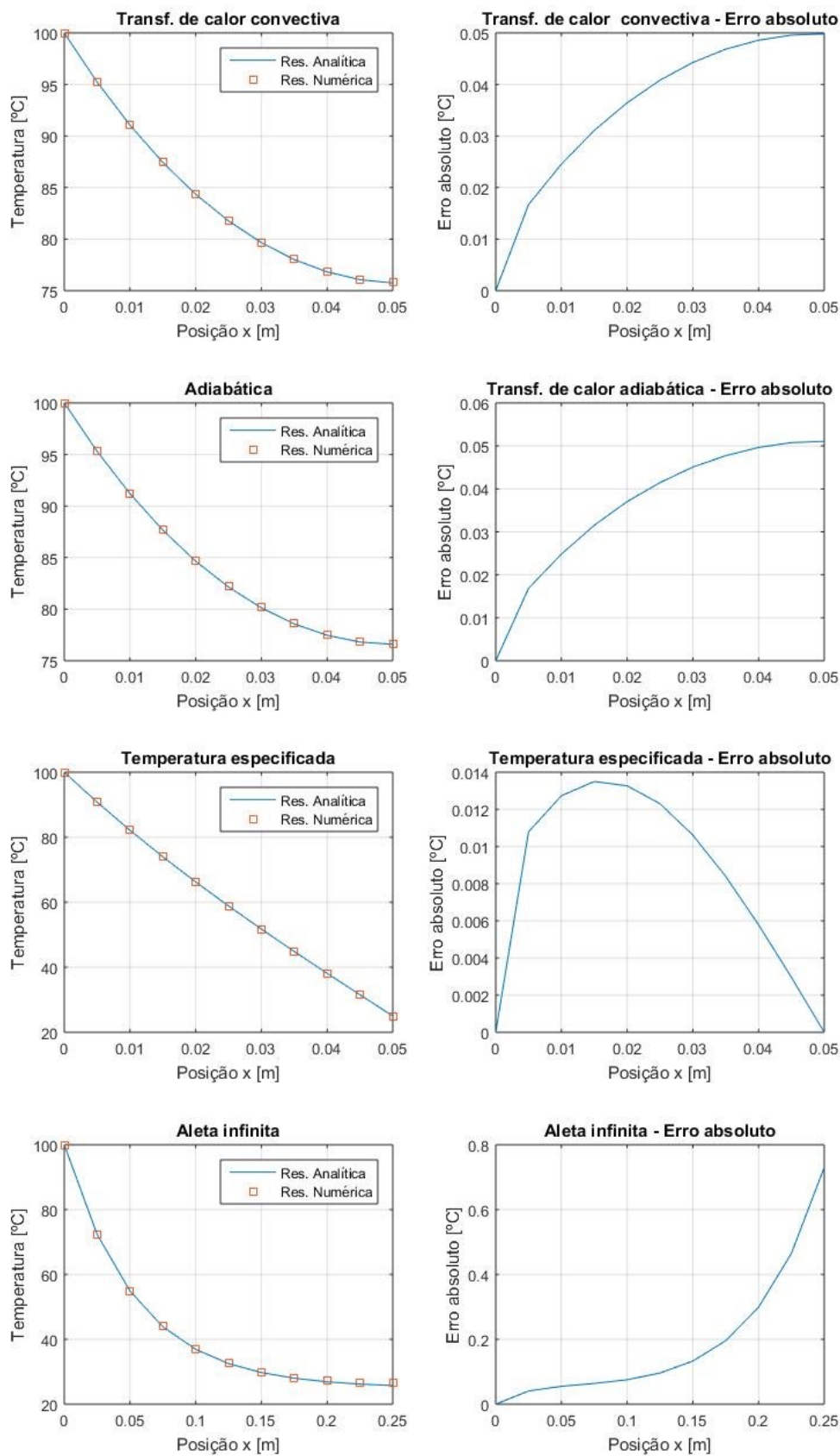
Figura 11 - Resultados do CTPP para teste de transferência de calor em aleta, caso A.



Fonte: Próprio autor.

O CTPP foi executado para cada caso com as condições descritas, em regime permanente. Os valores da temperatura de 11 pontos distribuídos ao longo do comprimento da aleta foram obtidos e sobrepostos à curva de resultados analíticos. Os resultados do teste 1 são mostrados na figura 12. Como pode ser visto na figura 12, a distribuição de temperaturas calculada pelo programa é praticamente igual à distribuição obtida das equações analíticas. Os gráficos de erro absoluto, à direita de cada gráfico de comparação de temperaturas, indicam a proximidade dos resultados. Nos casos (A) e (B) o erro permaneceu menor que 0,07% e no caso (C), o erro foi menor que 0,022%. No caso (D), o erro aumentou ao longo do comprimento da aleta, alcançando cerca de 0,7 °C, o que corresponde a um erro relativo de 2,8%. O maior erro no caso da aleta infinita se deve ao acúmulo de erros à medida que se aumenta a distância da extremidade com a base de temperatura pré-definida.

Figura 12 - Gráficos de comparação dos resultados do CTPP com soluções analíticas, em transferência de calor em aletas.

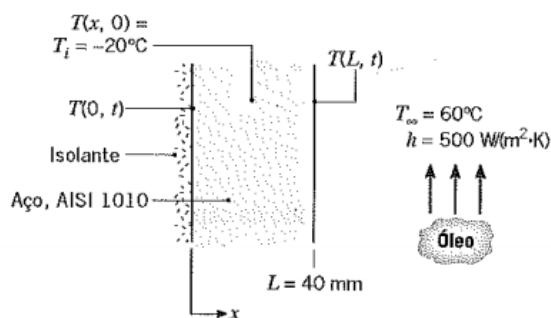


Fonte: Próprio autor.

4.2 TESTE 2

No segundo teste foi feita uma comparação dos resultados dados pelo CTPP com o resultado de parte do exemplo 5.4 de Incropera, página 175.

Figura 13 - Esquema do exemplo 5.4 de Incropera.



Fonte: Incropera (2008, p. 175).

Trata-se do aquecimento da parede de um duto. Neste caso, a parede do duto é considerada plana na solução analítica. A parede externa do duto é isolada e em seu interior escoia fluido a 60°C , causando uma transferência de calor convectiva com $h = 500 \text{ W}/(\text{m}^2 \cdot \text{K})$. A temperatura inicial da parede é de -20°C , e sua espessura é de 40 mm . Busca-se a temperatura da parede externa do duto após 8 min . No exemplo, a temperatura é calculada por meio da solução do primeiro termo da equação do calor. As seguintes propriedades foram consideradas:

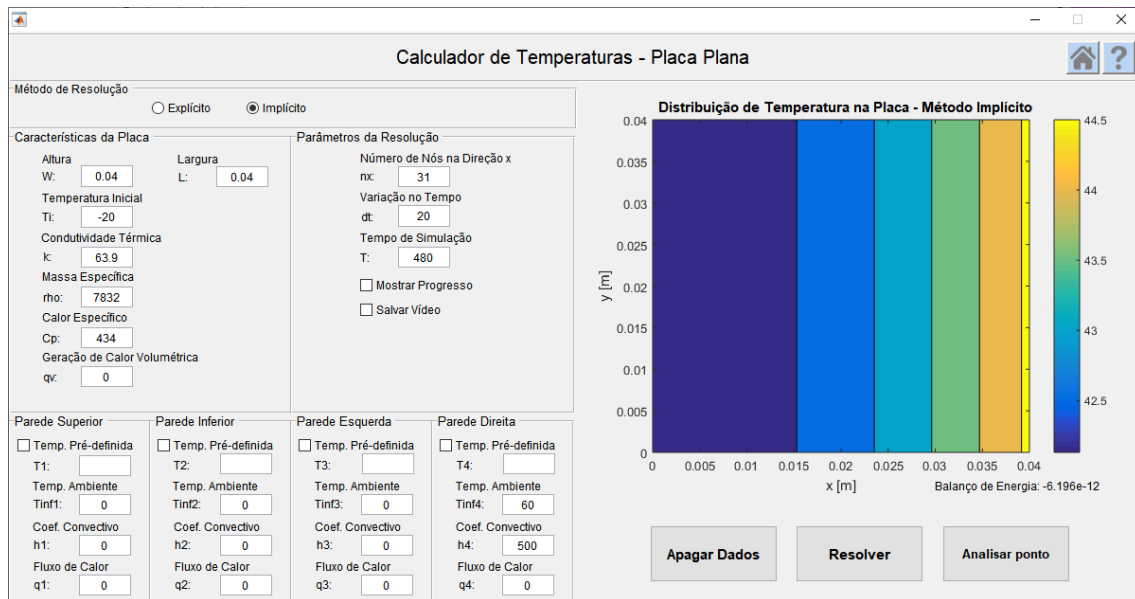
$$\rho = 7832 \text{ kg}/\text{m}^3$$

$$c = 434 \text{ J}/(\text{kg} \cdot \text{K})$$

$$k = 237 \text{ W}/(\text{m} \cdot \text{K})$$

A figura 14 exibe a interface do CTPP, mostrando os dados inseridos na solução e o resultado calculado pelo programa.

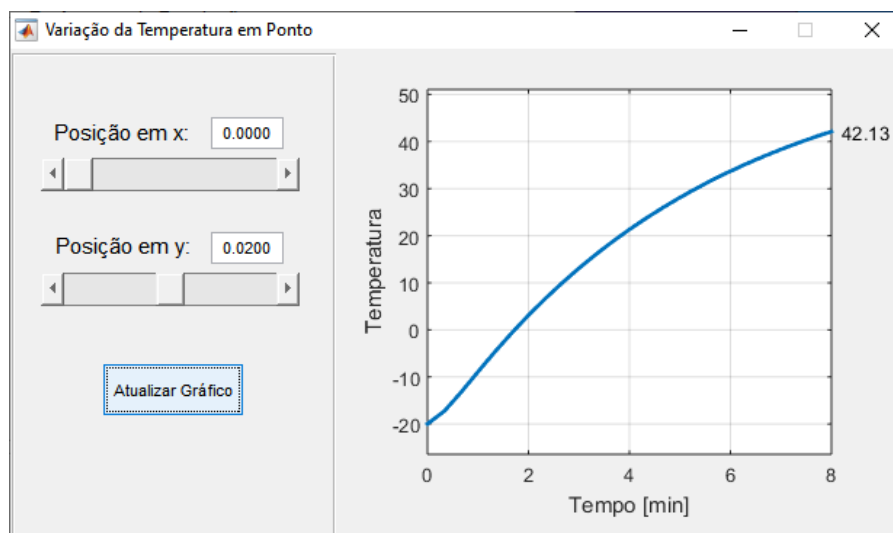
Figura 14 - Tela principal do CTPP em solução do teste 2.



Fonte: Próprio autor.

No exemplo, a temperatura externa calculada após 8 minutos é de $42,9^{\circ}\text{C}$. No CTPP, conforme mostrado na figura 15, a temperatura calculada pela solução numérica do software foi de $42,13^{\circ}\text{C}$. Houve, portanto, um erro relativo de 1,79%. Essa divergência pode ser resultado das aproximações tanto da solução analítica aproximada quanto da solução numérica.

Figura 15 - Análise de temperatura em ponto em solução do teste 2.



Fonte: Próprio autor.

4.3 TESTE 3

O terceiro teste consiste da comparação da distribuição de temperaturas do CTPP com um software que calcula a distribuição de temperaturas de placa plana analiticamente, desenvolvido pelo LTCME – Laboratório de Ensino e Pesquisa em Transferência de Calor: Modelagem e Experimento da FEMEC – UFU. Na situação avaliada há fluxo de calor de 10^5 W/m^2 nas paredes esquerda e superior da placa. As paredes inferior e direita se encontram isoladas. As condições da placa são descritas abaixo:

$$\rho = 7750 \text{ kg/m}^3$$

$$c = 480 \text{ J/(kg. K)}$$

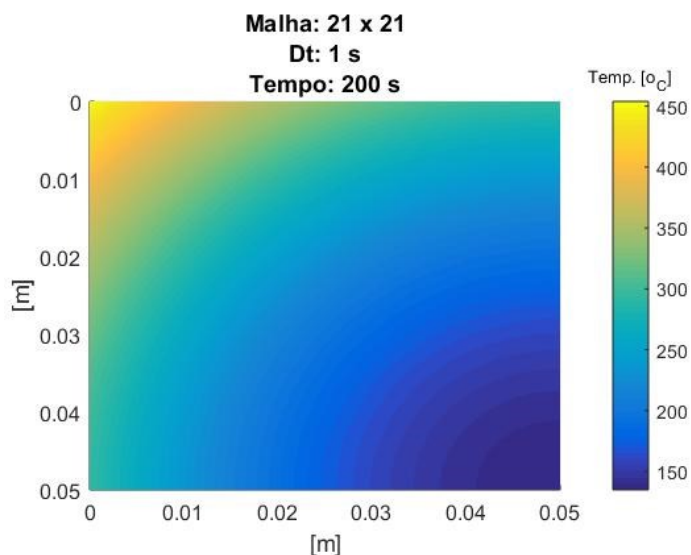
$$k = 15,1 \text{ W/(m. K)}$$

$$L = 0,05 \text{ m}$$

$$W = 0,05 \text{ m}$$

Na figura 16 é mostrada a distribuição de temperaturas após 200 segundos de transferência de calor, partindo de uma temperatura inicial de 30°C , calculada pelo software analítico.

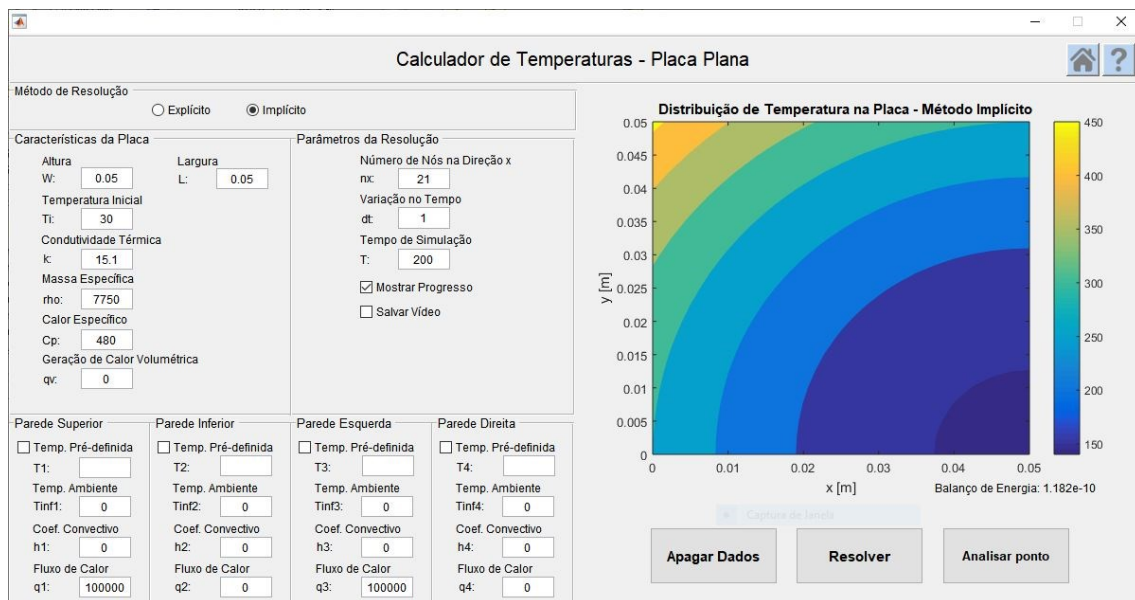
Figura 16 - Resultados de distribuição de temperatura em teste 3, solução analítica.



Fonte: Programa criado por Jefferson Gomes Do Nascimento, Universidade Federal de Uberlândia, 2017.

A mesma simulação foi realizada no CTPP. Os resultados são mostrados na figura 17. Pela comparação das figuras 16 e 17, o teste 3 também se mostra satisfatório, pois ambos métodos de solução forneceram resultados semelhantes. O formato da distribuição, o gradiente de cores e a faixa de temperatura obtidos após 200 segundos é visualmente igual nos dois softwares.

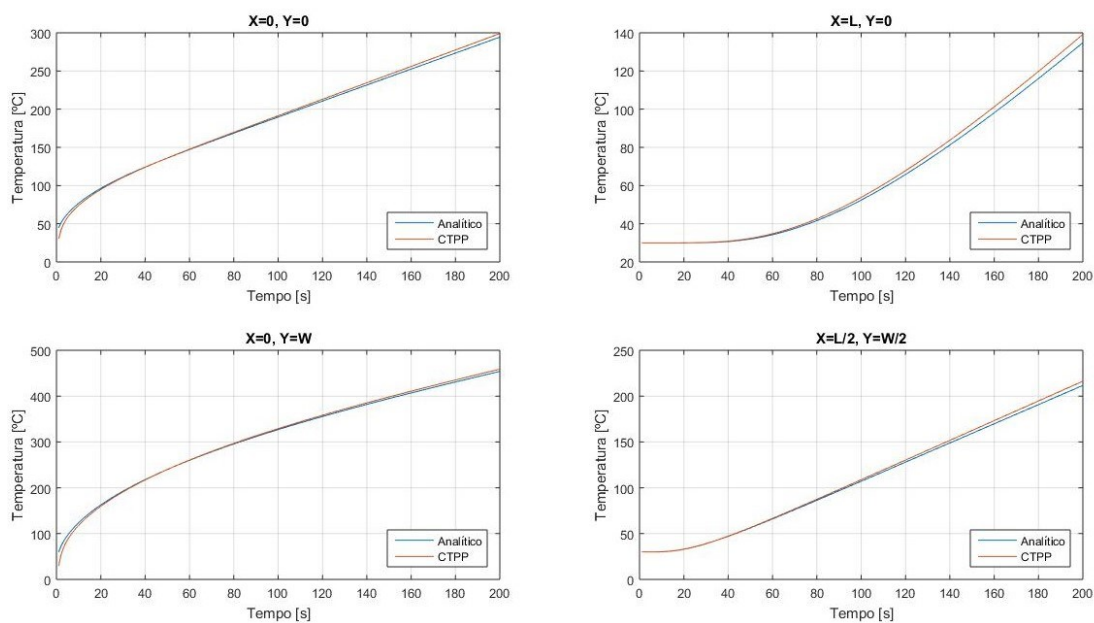
Figura 17 - Resultados da distribuição de temperaturas em teste 3, CTPP.



Fonte: Próprio autor.

A figura 18 mostra uma comparação dos resultados do software analítico e do CTPP, em pontos localizados nas extremidades e no centro da placa. O erro absoluto ao final da simulação variou entre 5,34°C e 6,08°C. O erro relativo percentual ficou entre 1,34% e 3,96%.

Figura 18 – Comparação entre software analítico e CTPP.



Fonte: Próprio autor.

Em resumo, três testes foram realizados, contemplando situações de transferência de calor unidimensional em regime permanente, unidimensional em regime transiente e bidimensional em regime transiente. Os resultados calculados pelo CTPP, em todos os casos, se mostraram coerentes com as resoluções analíticas.

5 CONCLUSÕES

Este projeto tinha como finalidade a elaboração de um software com uma interface de usuário amigável e intuitiva para cálculo da distribuição de temperaturas em problemas representados por placa plana bidimensional, através de solução numérica aplicando métodos de diferenças finitas explícito e implícito.

O software desenvolvido no projeto é capaz de fornecer distribuições de temperatura para tais problemas em diversas configurações comumente encontradas no ensino de transferência de calor na graduação. As configurações e características do problema podem ser facilmente inseridas pelo usuário através da interação com a interface sem necessidade de modificação nos códigos do programa. Os dados inseridos também são verificados a fim de evitar incoerências na solução e os erros encontrados são exibidos para que o usuário possa corrigi-los. A validade dos resultados pode ser verificada pelo usuário após cada execução, por meio de um balanço de energia exibido na interface. Também foram feitos testes de validação no presente trabalho, onde foi verificado que as soluções fornecidas pelo software são coerentes com soluções analíticas.

A distribuição do software é feita através de um pacote de instalação, não sendo necessário que o usuário possua o Matlab instalado em seu computador. O software é de fácil acesso e o uso da interface é simples, bastando que o usuário conheça os conceitos de transferência de calor abordados. Portanto, conclui-se que o objetivo inicial do projeto foi atendido e o software pode ser uma ferramenta útil no ensino de transferência de calor, capaz de proporcionar fácil interação com as variáveis presentes e assim estimular o aprendizado e interesse por essa área de conhecimento.

5.1 SUGESTÕES DE TRABALHOS FUTUROS

- Refazer a interface no *App Designer*, pois a ferramenta *GUIDE* será removida do *Matlab* futuramente.
- Refazer a interface em um software de livre acesso.
- Realizar testes de malha e outras comparações com software de solução analítica, como o desenvolvido no LTCME por Lauro Henrique (FRANÇA, 2021).
- Ampliar a interface para possibilitar solução de problemas tridimensionais.

BIBLIOGRAFIA

INCROPERA, Frank; DEWITT, David P.; BERGMAN, Theodore L.; LAVINE, Adrienne S. **Fundamentos de Transferência de Calor e Massa**. 6. ed. Rio de Janeiro, Brasil: LTC, 2008. 643 P.

ÇENGEL, Yunus A; GHAJAR, Afshin J. **Transferência de Calor e Massa: uma abordagem prática**. 4.ed. São Paulo, Brasil: AMGH Editora LTDA, 2012. 905 p.

BONNAND, Salomé Gabrielle Lucille. **Desenvolvimento de um modelo numérico 3D para técnica inversa de condução de calor**, 2017. 47 f. Projeto de Fim de Curso – Universidade Federal de Uberlândia, Uberlândia.

FRANÇA, Lauro Henrique Rodrigues De. **Desenvolvimento de um Software Amigável para a Análise de Soluções Analíticas em Problemas de Condução de Calor Transientes 1D, 2D e 3D**, 2021. 36 f. Projeto de Fim de Curso – Universidade Federal de Uberlândia, Uberlândia.

NASCIMENTO, Jefferson Gomes. **Desenvolvimento de uma plataforma numérica para soluções analíticas, híbridas e exatas de problemas diretos e inversos em condução de calor**. 2020. 36 f. Projeto de Fim de Curso - Universidade Federal de Uberlândia, Uberlândia.

MATHWORKS. **Guide Migration Strategies – MATLAB & Simulink**. 2021. Retirado de https://www.mathworks.com/help/matlab/creating_guis/differences-between-app-designer-and-guide.html, acesso: dezembro de 2021.