

INTEGRAÇÃO DE MÉTODOS DE MODELAÇÃO GEOMÉTRICA, ELEMENTOS FINITOS E PROGRAMAÇÃO MATEMÁTICA NO PROJECTO ESTRUTURAL

JOÃO MÁRIO BURGUETE BOTELHO CARDOSO *

e
CARLOS ALBERTO MOTA SOARES **

** Secção Autónoma Eng. Mecânica,
Faculdade Ciências e Tecnologia,
Quinta da Torre, 2825 Monte de Caparica, Portugal.*

*** Centro de Mecânica e Materiais da Universidade Técnica de Lisboa,
Instituto Superior Técnico,
Av. Rovisco Pais, 1096 Lisboa Codex, Portugal.*

SUMÁRIO

A optimização de forma de componentes estruturais, sujeitos a esforços de membrana, é obtida recorrendo a um modelo numérico que integra modelação geométrica, análise por elementos finitos e programação matemática. Curvas paramétricas são utilizadas para definir a fronteira do domínio, sendo a posição dos seus nós assumida como variáveis de projecto. Para cada configuração, um gerador de malha elíptico permite obter automaticamente a discretização elíptica necessária à solução por elementos finitos. As sensibilidades são calculadas pela técnica semi-analítica e pelo método directo, podendo ser visualizadas gráficamente. O método da programação sequencial convexa é utilizado na fase de optimização. A implementação é feita com ênfase na utilização de gráficos interactivos e em linguagens de programação normalizadas, de forma a permitir a portabilidade do sistema desenvolvido.

SUMMARY

Shape optimization of two dimensional elasticity problems is obtained with a numerical model which integrates geometric modelling, finite elements analysis and mathematical programming. Parametric curves are used to define the boundary of the domain, the position of their nodes is taken as design variables. For each configuration, an elliptic mesh generator obtains automatically the finite element discretization necessary to get the solution. Sensibilities are calculated by the semi-analytical technique and by the direct method, and can be visualized graphically. The method of sequential convex programming is used at the optimization phase. Implementation is made with emphasis on interactive graphics and in standard computer languages, allowing portability of the developed system.

Recibido: Octubre 1988

INTRODUÇÃO

Assistiu-se nos últimos anos a consideráveis progressos no domínio da optimização estrutural. Muito do esforço actual de investigação é dirigido para o desenvolvimento de sistemas de fácil utilização, capazes de resolver uma grande variedade de problemas práticos.

Para isso pretende-se integrar os métodos de modelação e a utilização de gráficos interativos dos modernos sistemas de desenho assistido por computador, o método de análise por elementos finitos e os métodos de programação matemática e cálculo de sensibilidades da optimização estrutural.

No contexto da optimização de forma, essa integração apresenta contudo algumas dificuldades.

- O modelo geométrico, definido inicialmente no programa de desenho assistido por computador, é modificado quando se procede á optimização, pelo que se torna necessário uma actualização desse modelo durante o processo. Para que a integração seja completa, é necessário que o optimizador possa agir directamente sobre o modelo geométrico armazenado na base de dados do programa de desenho assistido por computador.
- As variações de forma ocorridas fazem depender o processo da optimização da utilização de geradores automáticos de malha eficazes. É necessário que os métodos utilizados possam, sem a intervenção do utilizador, gerar malhas com uma grande regularidade, mesmo para geometrias diferindo por vezes consideravelmente da inicial. A obtenção do erro de aproximação cometido em cada análise de elementos finitos pode dar indicações úteis sobre a validade dos resultados obtidos, ou ser usado para controlar um processo adaptativo.

A abordagem realizada por vários autores, como Fleury e Liefoghe¹, Esping² e Rasmussen³, veio colocar a questão de saber se, para conseguir os objectivos propostos, se poderiam utilizar programas já existentes de desenho assistido por computador ou de elementos finitos, beneficiando do esforço de investigação efectuado para o desenvolvimento de muitos deles. A questão tem respostas diferentes nos dois casos. Enquanto a utilização de programas de análise por elementos finitos não apresenta grandes dificuldades, bastando, para poderem ser utilizados, que tenham formatos de entrada de dados e saída de resultados bem definidos, a utilização de programas de desenho assistido por computador levanta obstáculos, por não ser possível conseguir a estreita ligação que deve existir com o módulo de optimização. Contudo, a existência de linguagens de programação, ou linguagens de macros, na maioria destes sistemas, pode permitir ultrapassar este problema.

Neste trabalho apresenta-se um modelo numérico desenvolvido para a optimização de forma de componentes estruturais no domínio elástico, nos quais é válida a hipótese de tensão plana, que integra definição do modelo geométrico e do modelo de projecto, geração automática de malha, análise por elementos finitos com estimativa do erro de aproximação da solução, cálculo de sensibilidades e programação matemática.

A fronteira do domínio a analisar, definida por splines cúbicas e segmentos de recta,

é controlada por alguns pontos. O módulo de optimização move esses pontos de forma a obter a configuração óptima. A geração de malha e a análise por elementos finitos é feita automaticamente para cada nova configuração.

Apresentar-se o modelo geométrico, baseado no conceito de elemento de projecto e em splines paramétricas. A criação deste modelo, a definição das forças aplicadas e condições fronteira, a indicação das variáveis de projecto é feita utilizando gráficos interactivos com recurso a periféricos próprios – monitores de alta resolução e dispositivos apontadores.

Introduz-se o gerador de malha elíptico onde um sistema de equações de Poisson, juntamente com condições fronteira de Neumann, define numericamente uma transformação de coordenadas que estabelece a posição dos nós da malha. Esta forma de gerar a malha, ao contrário do método algébrico utilizado por vários autores^{1,2,3}, permite manter uma grande regularidade quando o domínio do problema sofre alterações de forma importantes. Contudo só uma referência apresenta a sua aplicação á optimização de forma⁴.

A formulação numérica é desenvolvida utilizando o elemento finito isoparamétrico de quatro nós para tensão plana⁵, sendo estimado o erro de aproximação no cálculo das tensões através do método indicado por Zienkiewicz e Zhu⁶. Este método permite obter também tensões projectadas nos nós dos elementos, a partir dos valores calculados nos pontos da Gauss, que são empregues nos gráficos de contorno de tensões.

O problema de programação matemática é resolvido pelo método da programação sequencial convexa desenvolvido por Fleury⁷, sendo utilizada a implementação do programa ADS-Automated Design Synthesis⁸. Este método requer o cálculo dos gradientes da função objectivo e dos constrangimentos, sendo estes calculados pelo método directo, utilizando diferenças finitas para derivar a matriz de rigidez de cada elemento. São apresentados resultados, demonstrativos da eficácia do modelo utilizado.

MODELAÇÃO

O modelo numérico de um elemento estrutural deve conter a informação necessária para uma descrição das características relevantes desse elemento. Por conveniência, podemos decompor o modelo aplicado na resolução do problema de optimização de forma em vários sub-modelos, de acordo com a informação acumulada em cada um:

Modelo Geométrico — Contém uma descrição da geometria do problema.

Modelo Físico — Contém informação sobre as características físicas, como propriedades dos materiais, forças aplicadas e deslocamentos restringidos, correspondentes á configuração existente num determinado instante. Inclui também a geometria, pelo que este modelo engloba o modelo geométrico.

Modelo de Elementos Finitos — Contém informação semelhante ao modelo físico, mas adaptada á utilização do método dos elementos finitos, isto é, o domínio discretizado em elementos, as forças aplicadas nos nós, etc.

Modelo de Projecto — Contém a informação sobre o problema de optimização de forma, como variáveis e constrangimentos e inclui o modelo físico.

Para definir a geometria é necessário traçar as linhas que assinalam a fronteira do domínio. Pelas sua regularidade característica, são aqui utilizadas splines cúbicas naturais, embora outras curvas paramétricas tenham sido já aplicadas com sucesso em optimização de forma, como splines de Ferguson² e B-splines³. É ainda possível utilizar curvas de Bezier ou simplesmente arcos de círculo ou segmentos de recta. O sistema está construído de forma a permitir a adição de todos estes tipos de curvas, estando já implementadas as duas últimas referidas.

Baseado nestas linhas define-se o modelo geométrico, que é constituído por um ou mais blocos, sendo cada um destes um conjunto de linhas. Cada conjunto é ordenado pois o início de cada linha, excepto a primeira, corresponde ao fim da anterior e o fim da última linha corresponde ao início da primeira.

O modelo geométrico pode ser construído de raiz ou a partir de linhas pertencentes a um desenho que foram importadas de outro sistema. No primeiro caso, é necessário traçá-las, assinalando as coordenadas dos nós de cada uma, através do teclado ou apontando directamente no monitor por intermédio de um dispositivo adaptado ao efeito, e em seguida formar com elas sub-domínios fechados ou blocos. Para cada um destes, indicam-se as linhas que o constituem, sequencialmente e no sentido directo, adicionando informação suplementar destinada ao gerador de malha. No segundo caso apenas esta última fase é necessária.

Para obter o modelo físico completo, acrescenta-se o índice que identifica as propriedades do material utilizado, e que constam de um ficheiro, as forças aplicadas e as restrições ao movimento. Para definir forças e restrições basta indicar a linha que delimita a fronteira onde estas condições estão aplicadas, juntamente com a valor das forças ou os graus de liberdade restringidos. Neste processo, semelhante ao utilizado em⁹, as condições fronteira são associadas a linhas, o que torna possível, por exemplo, a sua definição antes da malha de elementos finitos ser gerada ou mesmo antes do domínio estar delimitado na sua totalidade.

O modelo de projecto contém, para além da anterior, toda a informação relativa a variáveis de projecto e a constrangimentos, pois é sobre este modelo que actua o módulo de optimização. Para cada posição das variáveis, o sistema cria um modelo físico e a partir deste, automaticamente, um modelo de elementos finitos.

GERAÇÃO AUTOMÁTICA DE MALHA

Para gerar a malha necessária á criação do modelo de elementos finitos, o sistema recorre a um gerador de malha elíptico com condição de ortogonalidade na fronteira.

O gerador utiliza um sistema de duas equações diferenciais elípticas para definir uma transformação de coordenadas entre o domínio Ω do problema, definido no referencial X, Y e com uma forma arbitrária, e um domínio rectangular Ω^* , definido no referencial ρ, η . A transformação é feita de forma a fazer corresponder cada um dos troços da fronteira do domínio $\Omega, \Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4$ respectivamente os troços $\Gamma_1^*, \Gamma_2^*, \Gamma_3^*, \Gamma_4^*$ do domínio Ω^* .

São utilizadas as equações de Poisson:

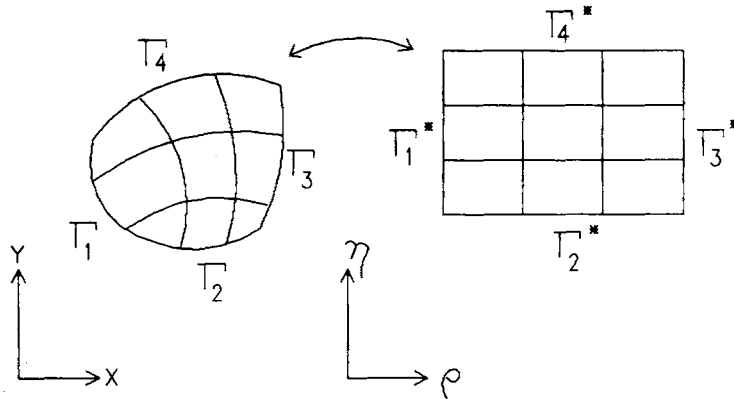


Figura 1. Transformação definida pelas equações elípticas.

$$\begin{aligned} \rho_{,XX} + \rho_{,YY} &= P(\rho, \eta) \\ \eta_{,XX} + \eta_{,YY} &= Q(\rho, \eta) \end{aligned} \tag{1}$$

com condições fronteira:

$$\begin{aligned} \rho &= \rho_1 && \text{para } (X, Y) \in \Gamma_1 \\ \rho &= \rho_2 && \text{para } (X, Y) \in \Gamma_2 \\ \eta &= \eta_1 && \text{para } (X, Y) \in \Gamma_3 \\ \eta &= \eta_2 && \text{para } (X, Y) \in \Gamma_4 \end{aligned} \tag{2}$$

As funções $P(\rho, \eta)$ e $Q(\rho, \eta)$ permitem, como demonstrado por Thompson¹⁰, modificar a distribuição das linhas que, no domínio Ω , correspondem a ρ e η constantes. Estas linhas, devido á regularidade das soluções da equação de Poisson, distribuem-se de forma regular no domínio e são por isso utilizáveis para definir uma malha de elementos finitos. Se as funções anteriores forem iguais a zero em todo o domínio, então obtemos um sistema de equações de Laplace que garante a ortogonalidade das linhas mencionadas, nesse domínio, o que se traduz na eficiência das malhas geradas.

Para que a malha apresente ortogonalidade na fronteira, é necessário introduzir as condições adicionais:

$$\begin{aligned} \nabla \eta \cdot \mathbf{n} &= 0 && \text{para } (X, Y) \in \Gamma_1 \\ &&& \text{e } (X, Y) \in \Gamma_3 \\ \nabla \rho \cdot \mathbf{n} &= 0 && \text{para } (X, Y) \in \Gamma_2 \\ &&& \text{e } (X, Y) \in \Gamma_4 \end{aligned} \tag{3}$$

As equações (1) são resolvidas numericamente no domínio regular Ω^* , para o que são primeiro invertidas, isto é, passando as coordenadas X, Y a ser função de ρ e η .

O sistema de equações diferenciais elípticas resultante, apesar de mais complexo que o original, pode ser facilmente resolvido utilizando diferenças finitas, devido á

regularidade do domínio Ω^* , sendo a solução do sistema não-linear originado por estas obtida pelo método da sobre-relaxação.

CÁLCULO DE SENSIBILIDADES

Em problemas de elasticidade linear, a utilização do método dos elementos finitos conduz á solução do sistema:

$$\mathbf{K} \mathbf{u} = \mathbf{f} \quad (4)$$

onde \mathbf{K} é a matriz de rigidez, \mathbf{u} o vector dos deslocamentos nos nós e \mathbf{f} o vector contendo as forças nodais. As tensões em cada elemento, σ são obtidas a partir desses deslocamentos através da matriz \mathbf{T} :

$$\sigma = \mathbf{T} \mathbf{u} \quad (5)$$

\mathbf{K} , \mathbf{f} e \mathbf{T} dependem normalmente das variáveis de projecto X .

Os métodos de programação matemática implementados no programa ADS necessitam de conhecer o gradiente da função objectivo, neste caso a área, e ainda os gradientes dos constrangimentos em ordem ás variáveis de projecto anteriormente apresentadas. Podendo considerar constrangimentos no deslocamento ou na tensão em cada nó do tipo:

$$\psi_{\mathbf{u}_j} = \mathbf{u}_j - \bar{\mathbf{u}} \leq 0 \quad , \quad \psi_{\sigma_j} = \sigma_j - \bar{\sigma} \leq 0 \quad (6)$$

sendo \mathbf{u}_j , σ_j um deslocamento e uma componente da tensão no nó j , esses gradientes serão dados por:

$$\text{grad } \psi_{\mathbf{u}_j} = \mathbf{u}_{j,X_1} \hat{e}_1 + \mathbf{u}_{j,X_2} \hat{e}_2 + \dots + \mathbf{u}_{j,X_n} \hat{e}_n \quad (7)$$

$$\text{grad } \psi_{\sigma_j} = \sigma_{j,X_1} \hat{e}_1 + \sigma_{j,X_2} \hat{e}_2 + \dots + \sigma_{j,X_n} \hat{e}_n \quad (8)$$

onde $\hat{e}_1, \hat{e}_2, \dots, \hat{e}_n$ são os versores do espaço das variáveis de projecto.

Para obter as derivadas ou sensibilidades de \mathbf{u}_j ou σ_j em relação a uma variável X_i , \mathbf{u}_{j,X_i} ou σ_{j,X_i} , é sempre possível resolver as equações (4), (5) para valores próximos X_i e $X_i + \delta X_i$, e depois fazer:

$$u_{j,X_i} = \frac{u_j(X_i + \delta X_i) - u_j(X_i)}{\delta X_i} \quad , \quad \sigma_{j,X_i} = \frac{\sigma_j(X_i + \delta X_i) - \sigma_j(X_i)}{\delta X_i} \quad (9)$$

esta técnica de diferenças finitas implica duas análises completas por cada variável de projecto e é muito dispendiosa e instável numericamente. Alternativamente as sensibilidades dos deslocamentos e tensões podem ser obtidas diferenciando as equações (4)(5):

$$\mathbf{K} \mathbf{u}_{,X_i} + \mathbf{K}_{,X_i} \mathbf{u} = \mathbf{f}_{,X_i} \quad (10)$$

$$\sigma_{,X_i} = \mathbf{T}_{,X_i} \mathbf{u} + \mathbf{T} u_{,X_i} \quad (11)$$

A equação (10) pode ser utilizada para obter as sensibilidades dos deslocamentos:

$$\mathbf{u}_{,X_i} = \mathbf{K}^{-1}(\mathbf{f}_{,X_i} - \mathbf{K}_{,X_i} \mathbf{u}) \quad (12)$$

o que equivale a resolver novamente a equação (4) para um novo sistema de forças fictícias \mathbf{q} igual a:

$$\mathbf{q} = \mathbf{f}_{,X_i} - \mathbf{K}_{,X_i} \mathbf{u} \quad (13)$$

A partir das sensibilidades nos deslocamentos calculam-se as sensibilidades nas tensões através de (11). \mathbf{u}_{j,X_i} , σ_{j,X_i} são os valores referentes ao nó j de $\mathbf{u}_{,X_i}$, $\sigma_{,X_i}$.

De notar que u_{j,X_i} , σ_{j,X_i} também podem ser obtidas a partir das equações (10), (11) calculando as variáveis adjuntas λ_{u_j} , λ_{σ_j} :

$$\lambda_{u_j} = \mathbf{K}^{-1} \mathbf{b} \quad (14)$$

$$\lambda_{\sigma_j} = \mathbf{K}^{-1} \mathbf{t} \quad (15)$$

onde \mathbf{b} é um vector que identifica a componente do deslocamento considerada, isto é, $u_j = \mathbf{b}^T \mathbf{u}$, e \mathbf{t} uma coluna do vector T tal que $\sigma_j = \mathbf{t}^T \mathbf{u}$, e em seguida fazendo:

$$u_{j,X_i} = \lambda_{u_j}(\mathbf{f}_{,X_i} - \mathbf{K}_{,X_i} \mathbf{u}) \quad (16)$$

$$\sigma_{j,X_i} = \mathbf{t}_{,X_i} \mathbf{u} + \lambda_{\sigma_j}(\mathbf{f}_{,X_i} - \mathbf{K}_{,X_i} \mathbf{u}) \quad (17)$$

O primeiro método, designado método directo, conduz á resolução da equação (12) tantas vezes quantas ás variáveis de projecto, enquanto o segundo método, o método adjunto obriga á solução de (14) ou (15) tantas vezes quantos os constrangimentos activos.

No modelo desenvolvido, as sensibilidades são calculadas pelo método directo pois este método tem a vantagem de, para cada variável, obter as sensibilidades em todos os nós da malha de elementos finitos, o que permite traçar gráficos como os que se apresentam. Estes gráficos, onde cada sensibilidade é tratada como variável de campo, fornecem ao utilizador indicações que o podem guiar na escolha da disposição óptima das variáveis de projecto, permitindo melhorar o projecto inicial, sem ter que recorrer á programação matemática.

O volume de cálculo efectuado por este método é proporcional ao número de variáveis de projecto, o que neste caso não constitui um inconveniente pois, devido á parametrização da fronteira utilizada, o número de variáveis é sempre reduzido. Por outro lado a implementação é feita de forma a diminuir esse volume de cálculo. Com

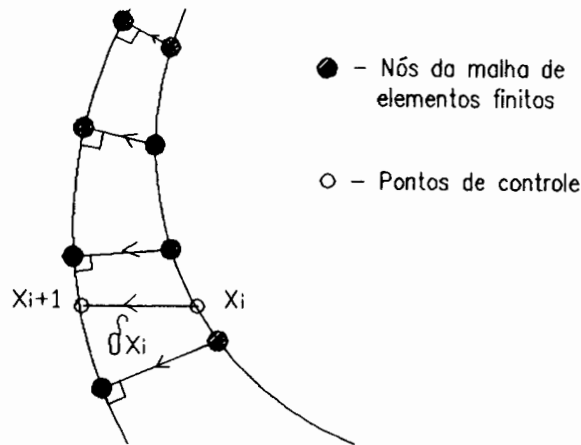


Figura 2. Modificação δX_i , na posição da variável i .

Numero de Elementos = 32
 Numero de Nós = 45
 Numero de Elementos afectados (Nel^*) = 8

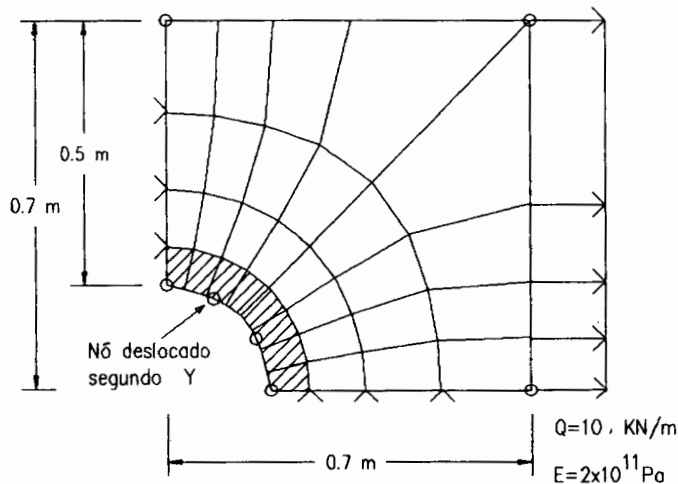


Figura 3. Exemplo placa com furo.

efeito para cada sensibilidade é necessário em primeiro lugar definir a posição dos nós correspondente á modificação δX_i na posição da variável i .

Só são afectados por esta modificação os nós sobre a fronteira a que pertence a variável i , que sofrem um deslocamento perpendicular á linha que define a nova posição dessa fronteira, como indicado na figura. Considerando na equação (13) que $f_{,X_i} = 0$, o vector de forças fictícias q é calculado a partir da contribuição dos elementos $n = 1 \dots Nel^*$, em cuja topologia exista um nó afectado, utilizando diferenças finitas para obter a derivada da matriz de rigidez desse elemento, isto é:

$$q = \sum_{n=1}^{Nel^*} \frac{K_e(X_i) - K_e(X_i + \delta X_i)}{\delta X_i} u_e \quad (18)$$

No	Sensibilidade semi-analitica	Sensibilidade por diferenças finitas globais	% Erro
1	—	—	—
2	-0.46026	-0.46345	-0.6
3	+0.07027	+0.08147	-13.7
4	+0.88601	+0.90970	-2.6
5	+1.29510	+1.32160	-2.0
6	+1.34970	+1.37580	-1.9
7	+1.21630	+1.24010	-1.9
8	+1.16960	+1.19480	-2.1
9	+1.16820	+1.19030	-1.9

Quadro I. Valores da sensibilidade do deslocamento segundo $X(x10^{-6})$.

No	Sensibilidade semi-analitica	Sensibilidade por diferenças finitas globais	% Erro
1	-3.02030	-3.041398	-0.7
2	+0.19414	+0.221769	-12.5
3	+2.48540	+2.548076	-2.5
4	+0.44433	+0.470693	-5.6
5	-1.18160	-1.203887	-1.9
6	-1.10510	-1.115179	-0.9
7	-0.44987	-0.455305	-1.2
8	-0.21953	-0.230956	-4.9
9	-0.21992	-0.212717	+3.4

Quadro II Valores da sensibilidade da tensão σ_{xx} .

A solução da equação (4) pelo método da factorização $L U$ de Crout permite guardar a matriz factorizada, pelo que a solução de (12) com a mesma matriz, apenas exige uma substituição regressiva. Uma vez obtido $u_{,X_i}$, o cálculo das sensibilidades das tensões é feito utilizando (11), sendo $T_{,X_i}$ calculado também por diferenças finitas.

Um dos factores que afecta a precisão dos resultados obtidos pela técnica semi-analítica é a magnitude de δX_i , sendo o erro total cometido igual á soma do erro de arredondamento com o erro de truncadura devido á utilização de um só termo na série de Taylor aplicada para calcular a derivada. Estas duas parcelas são afectadas de forma

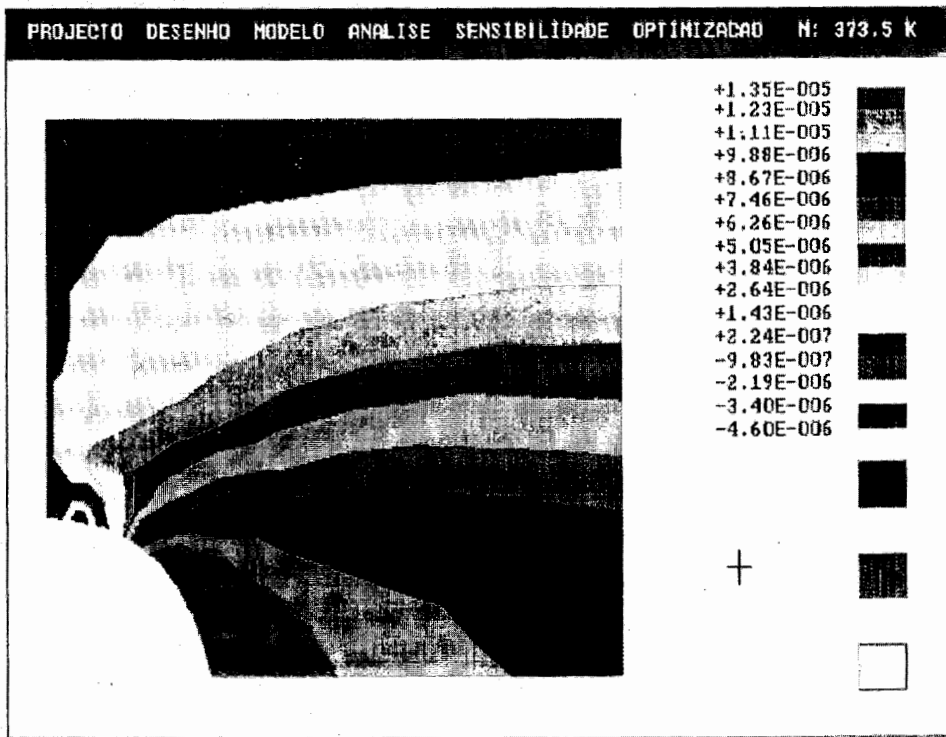


Figura 4. Sensibilidade do deslocamento segundo X.

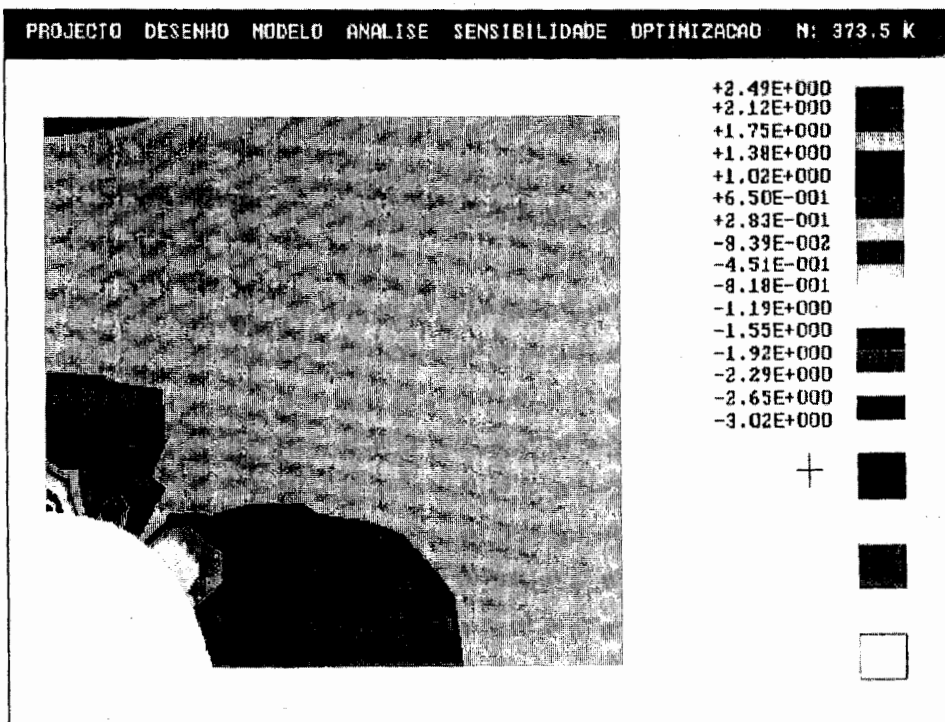


Figura 5. Sensibilidade da tensão.

diferente pela variação de δX_i , pois quando este aumenta o erro de arredondamento tende a diminuir e o erro de truncadura tende a aumentar, o que pode tornar difícil a escolha de um valor para δX_i conduzindo a resultados correctos.

Foram por isso aplicados vários valores de δX_i , calculados em função do comprimento dos elementos na direcção do movimento L , tendo-se verificado que os resultados não se alteravam significativamente para valores entre $0.01L$ e $0.06L$. Nas sensibilidades apresentadas seguidamente foi utilizado $\delta X_i = 0.029L$.

OPTIMIZAÇÃO DE FORMA

O problema de optimização é formulado como se indica:

$$\text{minimizar } F(X) \quad X = \{X_1, X_2, \dots, X_n\}^T \quad (19)$$

$$\text{com constrangimentos } \psi_j(X) \leq 0, \quad j = 1, 2, \dots, m \quad (20)$$

$$\text{e limites nas variáveis } \underline{X}_i \leq X_i \leq \bar{X}_i, \quad i = 1, 2, \dots, n \quad (21)$$

Em todos os exemplos apresentados foi utilizada a área do domínio como função objectivo. Como variáveis de projecto foram consideradas as posições dos pontos de controle, e como constrangimentos limites superiores no deslocamento ou na tensão em cada nó, pelo que o problema pode ser reformulado como:
minimizar a área:

$$A(X) = \sum_{n=1}^{Nel} A_n(X) \quad (22)$$

com constrangimentos

$$\begin{aligned} \psi_{u_j}(X) &= u_j - \bar{u} \leq 0, \quad j = 1, 2, \dots, m1 \\ \psi_{\sigma_j}(X) &= \sigma_j - \bar{\sigma} \leq 0, \quad j = 1, 2, \dots, m2 \end{aligned} \quad (23)$$

e limites nas variáveis de projecto

$$\underline{X}_i \leq X_i \leq \bar{X}_i, \quad i = 1, 2, \dots, n \quad (24)$$

Este é um problema de programação matemática não linear, com constrangimentos não lineares, podendo ser resolvido por vários métodos implementados no programa ADS. Foi utilizado o método proposto por Fleury e Braibant⁷ da programação sequencial convexa, que demonstrou ser eficaz na resolução deste tipo de problema³, permitindo obter em poucas iterações resultados muito próximos do óptimo.

Este método substitui o problema original por uma sequência de subproblemas aproximados, cada um dos quais é construído considerando a função objectivo e cada um dos constrangimentos aproximado por uma expansão em série de Taylor de 1ª ordem em torno do ponto do espaço de variáveis correspondente à iteração corrente X^o . Essa expansão é feita em relação às variáveis X_i ou em relação ao inverso dessas variáveis, $\frac{1}{X_i}$ de acordo com o sinal da derivada da função nesse ponto.

No ponto X^o é definida uma nova função e novos constrangimentos:

$$A^{\circ}(X) = A(X^{\circ}) + \sum_{+} \left(\frac{\partial A}{\partial X_i} \right) (X_i - X_i^{\circ}) + \sum_{-} \left(\frac{\partial A}{\partial X_i} \right) \frac{X_i^{\circ}}{\partial X_i} (X_i - X_i^{\circ}) \quad (25)$$

$$\psi_{u_j}^{\circ}(X) = \psi_{u_j}(X^{\circ}) + \sum_{+} \left(\frac{\partial \psi_{u_j}}{\partial X_i} \right) (X_i - X_i^{\circ}) + \sum_{-} \left(\frac{\partial \psi_{u_j}}{\partial X_i} \right) \frac{X_i^{\circ}}{\partial X_i} (X_i - X_i^{\circ}) \quad (26)$$

, $j = 1, 2, \dots, m1$

$$\psi_{\sigma_j}^{\circ}(X) = \psi_{\sigma_j}(X^{\circ}) + \sum_{+} \left(\frac{\partial \psi_{\sigma_j}}{\partial X_i} \right) (X_i - X_i^{\circ}) + \sum_{-} \left(\frac{\partial \psi_{\sigma_j}}{\partial X_i} \right) \frac{X_i^{\circ}}{\partial X_i} (X_i - X_i^{\circ}) \quad (27)$$

, $j = 1, 2, \dots, m2$

onde \sum_{+} e \sum_{-} significam somatório para todas as derivadas positivas e negativas, respectivamente. Prova-se que uma aproximação desta natureza é sempre convexa e conservativa, o que evita a utilização de limites ao movimento das variáveis X_i em cada iteração. Contudo, e em alguns casos estudados, verificou-se a necessidade de impor um limite de 20% para melhorar a convergência.

O subproblema criado para a iteração X° pode ser formulado como:
minimizar a área

$$A^{\circ}(X) \quad (28)$$

com constrangimentos

$$\begin{aligned} \psi_{u_j}^{\circ} &\leq 0 \quad , j = 1, 2, \dots, m1 \\ \psi_{\sigma_j}^{\circ} &\leq 0 \quad , j = 1, 2, \dots, m2 \end{aligned} \quad (29)$$

e limites nas variáveis de projecto

$$\underline{X}_i \leq X_i \leq \bar{X}_i \quad , \quad i = 1, 2, \dots, n \quad (30)$$

A obtenção do ponto óptimo em cada subproblema é conseguida pelo método das direcções admissíveis modificado¹¹, também implementado no programa ADS. Todos os valores necessários a esta fase da optimização são obtidos recorrendo á expansão em série, pelo que esse ponto óptimo é obtido muito rapidamente.

O cálculo da função objectivo e dos constrangimentos envolve, no problema de optimização de forma, uma análise de elementos finitos completa, incluindo geração de malha. Por isso é importante obter a solução óptima com o menor número de cálculos possível, o que é conseguido neste método, pois os cálculos referidos são efectuados apenas uma vez para criar cada subproblema.

APLICAÇÕES

São apresentadas três aplicações do modelo desenvolvido. O objectivo é minimizar a área, sendo imposto um limite para a tensão normal σ_{xx} no primeiro e para a tensão de Von Mises nos dois outros, como constrangimento em cada nó. No primeiro é obtida a forma óptima de uma viga flectida por uma carga concentrada na extremidade livre, sendo o resultado comparado com o valor teórico obtido pela teoria de vigas. No segundo, extensivamente referido na literatura^{1,2,3}, é obtida a configuração óptima de um furo numa placa submetida a um estado bi-axial de tensão. A convergência é rápida e o resultado obtido é semelhante ao apresentado por outros autores. Apenas um quarto da placa foi modelado. No terceiro exemplo é obtida a forma óptima de uma concordância entre secções diferentes, mas ao contrário da solução apresentada na literatura, é simultâneamente calculada a largura da secção mais estreita. Esta verifica o critério de dimensionamento para estados uniaxiais de tensão.

EXEMPLO: VIGA

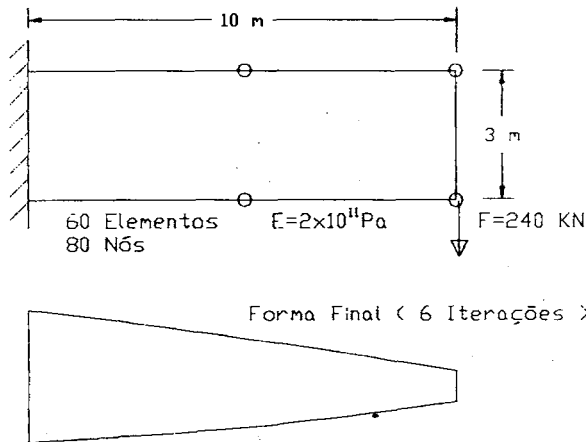


Figura 6. Exemplo viga.

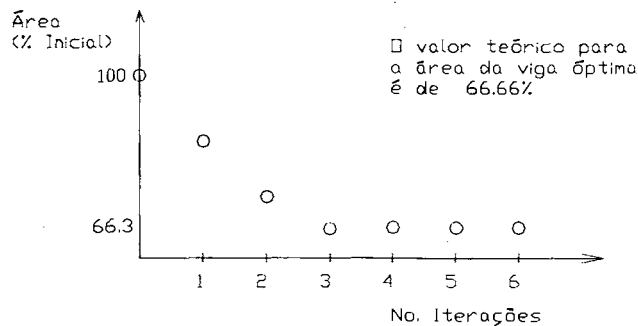
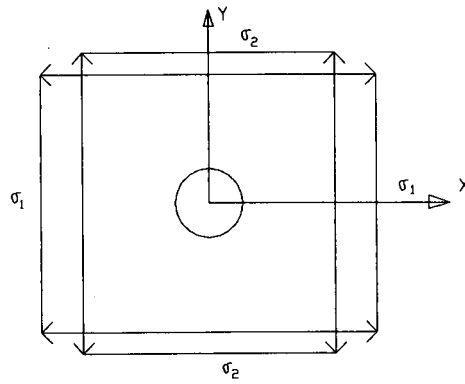


Figura 7. Variação da área durante a optimização.

EXEMPLO: PLACA COM FURO



Dimensões da placa
 1.4 x 1.4 m
 Diâmetro inicial do furo
 0.4 m
 $E = 2.0 \times 10^{11} \text{Pa}$
 $\sigma_1 = 2 \times \sigma_2 = 8 \text{ KN/cm}^2$

Modelo de Elementos Finitos
 (1/4 placa)
 48 Elementos
 63 Nós

Figura 8. Exemplo placa com furo.

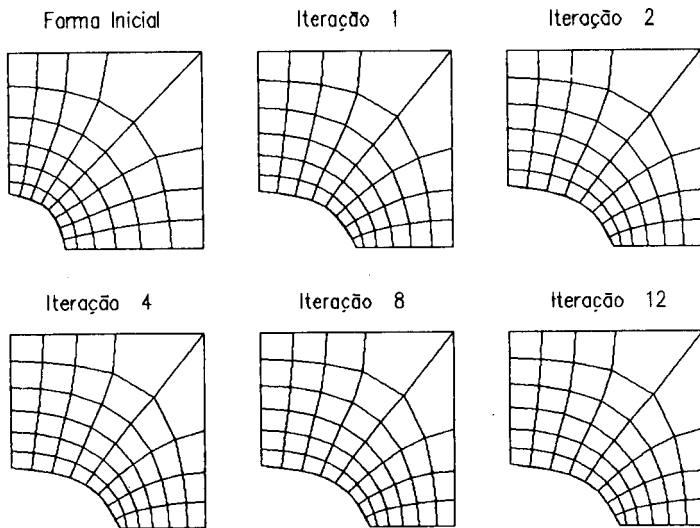


Figura 9. Discretização correspondente a várias iterações.

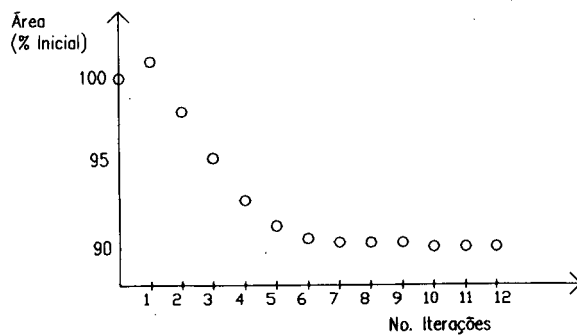


Figura 10. Variação da área durante a otimização.

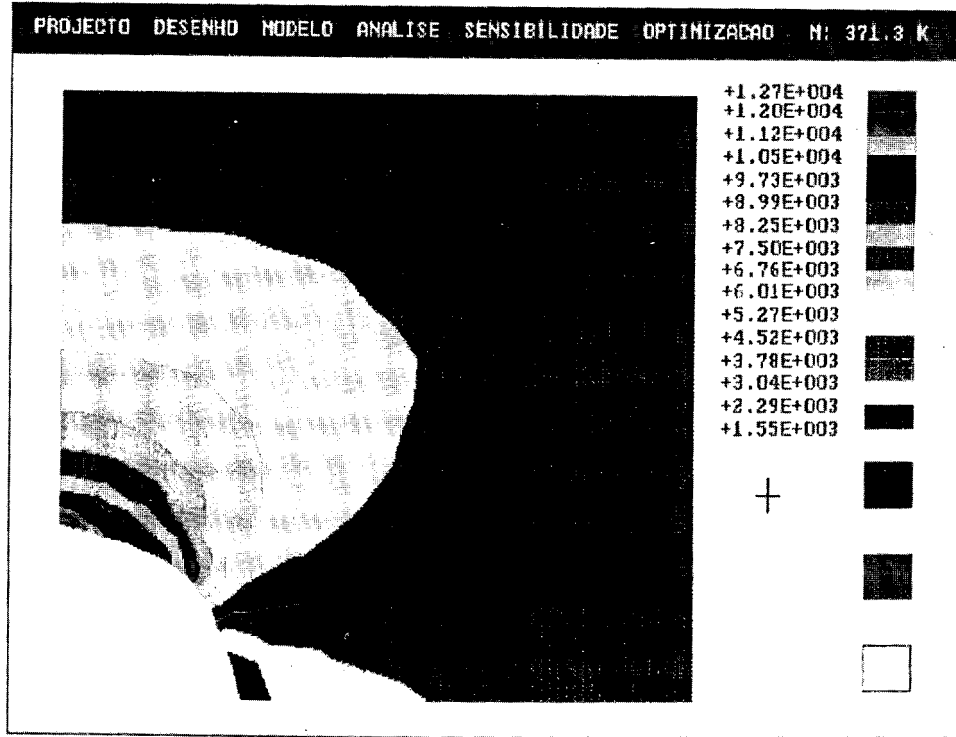


Figura 11. Distribuição inicial de Von Mises.

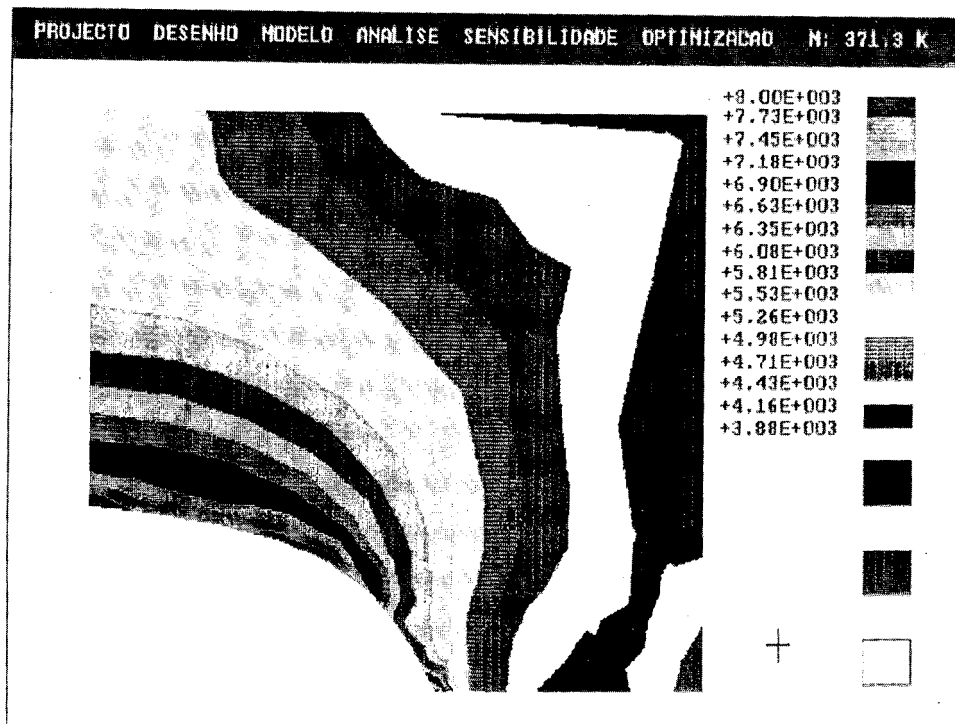


Figura 12. Distribuição final das tensões de Von Mises.

EXEMPLO: CONCORDÂNCIA

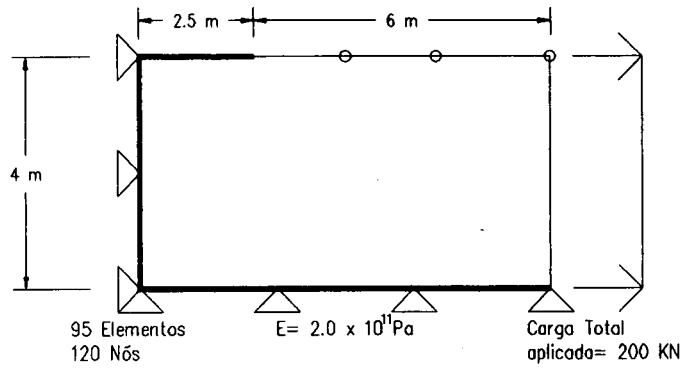


Figura 13. Exemplo concordância.

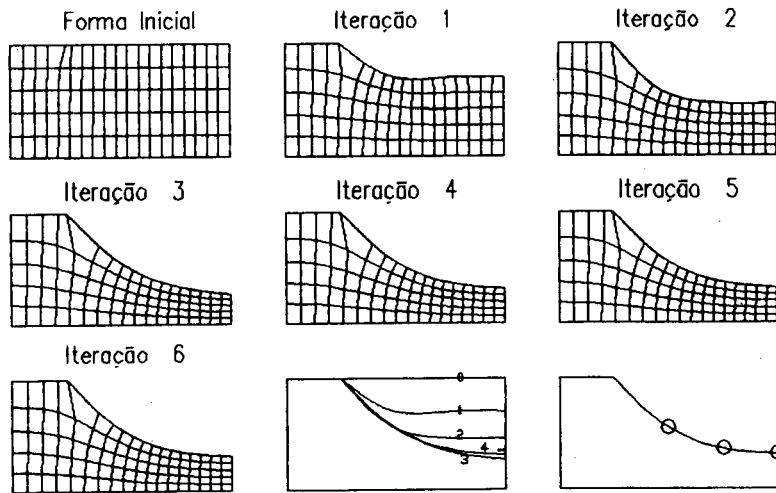


Figura 14. Discretização correspondente a várias iterações.

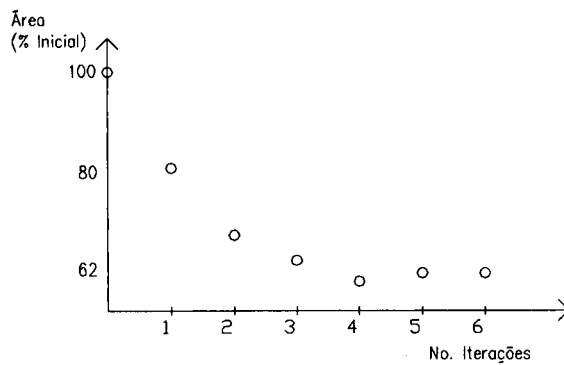


Figura 15. Variação da área durante a otimização.

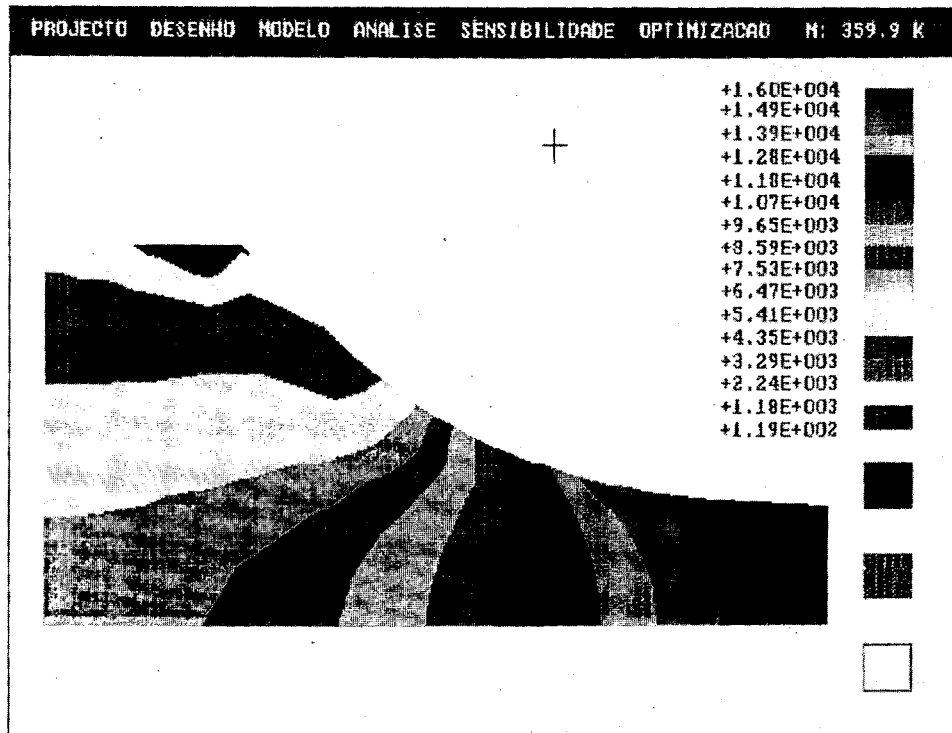


Figura 16. Distribuição da tensão σ_{xx} para a configuração final.

CONCLUSÕES

O método dos elementos finitos é hoje considerado um auxiliar precioso na análise do comportamento de estruturas e órgãos mecânicos, e com tal reveste-se de grande importância no projecto em engenharia.

Os métodos de modelação geométrica, surgidos recentemente na sequência do desenvolvimento das capacidades gráficas dos computadores actuais, juntamente com as técnicas de programação matemática foram aplicados com sucesso na actividade de projecto.

A utilização conjunta de todos estes métodos permite abordar esta actividade de uma forma nova, pelas possibilidades que oferece de resolver rápida e eficazmente problemas complexos. Contudo essa utilização conjunta no âmbito da optimização de forma dependente:

- 1- Da parametrização adequada do modelo geométrico. Para que o optimizador possa, agindo sobre esses parâmetros, gerar a forma óptima.
- 2- Da geração de malha automática e eficaz. Quando a geometria do problema se altera, o modelo de elementos finitos deve ser reconstruído, o que só é viável com geração automática. O gerador de malha deve produzir malhas regulares mesmo para geometrias que diferem por vezes consideravelmente da inicial.

No trabalho apresentado estes problemas foram resolvidos, obtendo-se uma perfeita

ligação entre modelação e optimização. O modelo geométrico é criado e depois constantemente actualizado durante o processo iterativo até atingir a forma óptima. O gerador de malha elíptico permite obter malhas caracterizadas pela sua regularidade, adaptando-se às variações de forma verificadas.

REFERÊNCIAS

1. C. Fleury e D. Liefoghe, "Shape optimal design on an engineering workstation", *MSC/N ASTRAN Users Conference*, Universal City, California, (1986).
2. B. Esping e D. Holm, "Structural shape optimization using OASIS", *IUTAM Symposium on Structural Optimization*, Melbourne, Australia, (1988).
3. J. Rasmussen, "Interactive shape optimization with design elements", *Report No. 14*, University of Aalborg, Institute of Mechanical Engineering, (1988).
4. H.C. Rodrigues, "Shape optimal design of elastic bodies using a mixed variational formulation", *Computer Methods in Applied Mechanics Engineering*, (1988).
5. N. Kikuchi, "*Finite Elements Methods in Mechanics*", Cambridge University Press, (1986).
6. O.C. Zienkiewicz e J.Z. Zhu, "A simple error estimator and adaptive procedure for practical engineering analysis", *International Journal for Numerical Methods in Engineering*, Vol. 24, pp. 337-357, (1987).
7. C. Fleury e V. Braibant, "Structural optimization: A new dual method using mixed variables", *International Journal for Numerical Methods in Engineering*, Vol. 23, pp. 409-428, (1986).
8. G.N. Vanderplaats, "*ADS - A Fortran Program for Automated Design Synthesis Version 2.01*", (1987).
9. J.A. Bennet, M.E. Botkin, "Structural shape optimization with geometric description and adaptive mesh refinement", *AIAA Journal*, Vol. 23, No. 3, (1985).
10. J.F. Thompson, Z.U.A. Warsi, C.W. Mastin, "*Numerical Grid Generation - Foundations and Applications*", North-Holland, (1985).
11. G.N. Vanderplaats, "*Numerical Optimizations Techniques for Engineering Design*", McGraw-Hill, (1984).