

**UNIVERSIDADE DE SÃO PAULO
ESCOLA POLITÉCNICA
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**Projeto de um novo tipo de espelho para telescópios
Um método para se obter espelhos asféricos de baixo
custo para telescópios**

André Izecson de Carvalho

Orientador: Emílio Carlos Nelli Silva

**São Paulo
2002**

UNIVERSIDADE DE SÃO PAULO
ESCOLA POLITÉCNICA
DEPARTAMENTO DE ENGENHARIA MECÂNICA

Projeto de um novo tipo de espelho para telescópios
Um método para se obter espelhos esféricos de baixo
custo para telescópios

Trabalho de formatura apresentado à
Escola Politécnica da Universidade
de São Paulo para a obtenção do
título de Graduação em Engenharia

André Izcson de Carvalho

Orientador:
Professor Doutor
Emílio Carlos Nelli Silva

Área de concentração:
Engenharia Mecânica

São Paulo
2002

**"Parece que a perfeição é atingida não no instante em que não há mais nada a acrescentar à máquina, mas no instante em que não há mais nada a suprimir.
Ao termo de sua evolução a máquina se dissimula."**

Saint-Exupéry

"A forma segue a função"

Louis Sullivan

Resumo:

Nas últimas décadas, o campo de astronomia amadora cresceu de apenas alguns pequenos grupos de entusiastas para uma indústria que movimenta milhões de dólares a cada ano. Há muitos tipos de telescópios disponíveis hoje para o astrônomo amador, desde pequenas e baratas lunetas, que usam lentes para formar a imagem, até grandes telescópios controlados por computador, que usam espelhos para formar a imagem. Esses espelhos são a parte mais importante de um telescópio. Eles definem a máxima resolução do telescópio, a luminosidade da imagem, e principalmente, seu preço.

Espelhos de telescópios são, geralmente, parabolóides ou hiperbolóides de revolução, e precisam ser usinados com uma precisão pelo menos igual a $1/10$ do comprimento de onda da luz, isto é, 50nm. A obtenção de superfícies com esta precisão exige um trabalho intenso e, portanto, é muito cara.

A manufatura de superfícies esféricas (parabolóides, hiperbolóides de revolução e etc.) é feita através de uma extensiva sequência de polimento manual, testes e correções do espelho, uma vez que essas superfícies não podem ser obtidas através de técnicas comuns de manufatura. Porém, isto não é o que acontece com superfícies esféricas, que são de obtenção relativamente fácil através do uso de máquinas polidoras simples que requerem pouca interferência humana. A precisão que se obtém na manufatura de superfícies esféricas é, em geral, muito maior do que o que se obtém com superfícies esféricas, não só por que superfícies esféricas são mais simples de se fabricar, como também por que são muito mais simples de se testar durante o processo de fabricação.

Assim, como é difícil de se obter superfícies esféricas com alta precisão, seria uma grande vantagem obter uma superfície esférica de alta precisão de deformá-la de forma a obter uma superfície esférica. Isso pouparia tempo e produziria um espelho de maior qualidade.

Baseado nesta idéia, este trabalho propõe um novo método de se obter superfícies de precisão para espelhos de telescópio. Elas podem se obtidas

pela aplicação de uma força no centro de um espelho esférico apoiado em três pontos de sua periferia. Se a superfície traseira do espelho tiver uma geometria adequada, a deformação resultante no espelho fará com que a superfície frontal se conforme em um parabolóide (ou qualquer outra superfície asférica desejada). A forma correta da superfície traseira do espelho pode ser determinada através de técnicas de otimização paramétrica. Para isso, o espelho foi modelado através de FEM, combinada com um algoritmo de otimização para encontrar a superfície traseira ótima. As variáveis de projeto utilizadas foram os coeficientes de uma superfície de Bèzier, que representava a superfície traseira do espelho. O algoritmo de otimização altera a forma da superfície através das iterações até que a superfície frontal esteja tão próxima quanto possível da superfície asférica desejada. Para a realização dos cálculos, o software comercial ANSYS foi usado.

Resultados preliminares mostraram que superfícies asféricas produzidas com esse método podem atingir uma precisão maior do que o que se obtém por métodos tradicionais, demonstrando que é possíveis se obter espelhos de alta precisão com um baixo custo.

Abstract:

In the last decades, the field of amateur astronomy has grown from essentially a few small groups of enthusiasts to a multi-million Dollar industry. There are many types of telescopes available today for amateur astronomers, from small, cheap refractors, that use lens to form an image, to large computer controlled telescopes that use mirrors to form the image. These mirrors are the most important component of a telescope. They define the maximum resolution of telescopes, image clarity, and ultimately, its price.

Telescope mirrors are usually a revolution paraboloid or hyperboloid, and must be machined with accuracy at least equal to $1/10$ of the wavelength of light, that is, 50nm. Obtaining surfaces of that level of accuracy is labor intensive and expensive.

The manufacturing of aspheric (revolution paraboloids, hyperboloids, etc) surfaces for small telescope mirrors is done through extensive hand-polishing, figuring and testing, once these surfaces cannot be obtained by usual manufacturing processes. However, it is not the case for spherical surfaces, which are relatively easily obtained by using simple polishing machines that barely require human assistance. The accuracy obtained in the manufacturing of spherical surfaces is usually much higher than what is obtained with aspherical surfaces, not only because the former is less complicated to manufacture, but also because it is much easier to test spherical surfaces during the manufacturing process.

Therefore, since it is hard to obtain aspherical surfaces with high accuracy, it would be a great advantage to be able to machine a high accuracy spherical surface and deform it to obtain an aspherical surface. This would save time and produce a mirror of superior quality.

Based on this idea, this work proposes a new method to obtain an accurate surface for telescope mirrors. This can be achieved by applying a force in the center of the spherical mirror supported through three points along its boundary. If the back surface is shaped properly, the resulting mirror deformation will cause the front spherical surface to conform into a paraboloid (or any other desired aspherical) surface. The correct shape of the mirror back

surface can be determined by using optimization techniques. To do this, the mirror was modeled and analyzed by FEM combined with an optimization algorithm to find the optimum back surface shape. The design variables of the optimization were the coefficients of Bezier surface which represents the back surface. The optimization algorithm changes the surface shape along the iterations until the front surface shape is as close as possible to the desired aspherical surface. For the calculations, the commercial software ANSYS was used.

Preliminary results showed that aspherical surfaces designed with this method could achieve greater accuracy than which is usually obtained by using traditional techniques demonstrating that it is possible to obtain high accuracy mirrors with low cost.

Índice

1.	Introdução	12
2.	Classificação de telescópios	14
2.1	Telescópios refratores	14
2.2	Telescópios Refletores	16
2.3	Telescópios Catadióptricos	19
3.	Os espelhos tensionados: Breve histórico	20
4.	Proposta para este trabalho	23
5.	Aspectos econômicos	24
6.	Definição das características desejadas do espelho	25
7.	Sobre a precisão necessária	30
8.	teste de foucault	33
9.	Aproveitamento da simetria	35
10.	Definição da função objetivo e restrições	36
11.	Resultados	38
11.1	Escolha de variáveis	38
11.1.1	Diferenças de cotas como variáveis	38
11.1.2	Cotas como variáveis	40
11.1.3	Coeficientes de um polinômio como variáveis	43
11.1.4	Curva de Bézier	44
11.2	Análise de sensibilidade	47
12	O Modelo tridimensional	48
12.1	Resultados	50
13	Comentários e conclusões	53
Anexo 1	Entrada do programa de otimização para o caso axisimétrico, com diferenças de cotas como variáveis	54
Anexo 2	Entrada do programa de otimização para o caso axisimétrico, com cotas como variáveis	61
Anexo 3	Entrada do programa de otimização para o caso axisimétrico, com coeficientes de um polinômio como variáveis	68
Anexo 4	Programa em MatLab que escreve a entrada do programa de otimização para o caso axisimétrico, com coeficientes de uma curva de Bézier como variáveis	76
Anexo 5	Programa em MatLab que escreve a entrada do programa de otimização para o caso tridimensional, com coeficientes de uma superfície de Bézier como variáveis	80
Anexo 6	Lista de fabricantes cujos telescópios foram adicionados no gráfico de Custos versus Abertura	91
	Crédito de imagens	92
	Bibliografia	93

Lista de figuras:

Nº	Legenda	página
1	Dois dos telescópios construídos por Galileu	12
2	Telescópio Refrator	14
3	Telescópio Refrator de Yerkes, o maior do mundo	15
4	Telescópio Refletor Newtoniano	16
5	O Telescópio de Isaac Newton	16
6	Esquema de telescópio Cassegrain ou Richey-Chrétien	17
7	Espelho do telescópio VLT, no Chile, com 8,2m	18
8	Telescópio VLT.	18
9	Telescópio Catadióptrico	19
10	Espelho esférico montado em uma célula fletora	20
11	Diagrama do espelho de Kelley	21
12	Protótipo do espelho de Kelley	21
13	Comparação entre a esfera e a parábola	25
14	Aberração esférica	26
15	Padrões de difração	30
16	Frente de onda	31
17	Esquema de montagem de um Teste de Foucault	33
18	Exemplo de teste de Foucault	34
19	Simetria do modelo	35
20	Diferença de cotas como variáveis	39
21	Resultados obtidos	39
22	Cotas como variáveis	41
23	Resultados obtidos	41
24	Coeficientes de um polinômio como variáveis	43
25	Resultados obtidos	44
26	Curva de Bézier	45
27	Corte do espelho com curva de Bézier	46
28	Corte do espelho obtido (eixo de simetria à esquerda)	46
29	Sistema de coordenadas	49
30	Superfície de Bézier	50
32	Resultado da otimização	51
33	Geometria obtida	52

Lista de gráficos:

Nº	Legenda	página
1	Preço de telescópios versus abertura	24
2	Aberração esférica	26
3	Deformação necessária	27
4	Planilhas de resultados	40
5	Planilhas de resultados	42
6	Planilhas de resultados	44
7	Planilhas de resultados	46
8	Planilhas de resultados	47

Lista de símbolos:

r	raio
$P(r)$	Parábola
f_p	Distância focal
$E(r)$	Esfera
$E'(r)$	Derivada da esfera no ponto r
$P'(r)$	Derivada da parábola no ponto r
R_c	Raio de curvatura da esfera
R	Raio do espelho
Y	Altura em que um raio de luz cruza o eixo ótico
$S(r)$	Uma curva qualquer
ρ	Diâmetro do disco de Airy
λ	Comprimento de onda de luz
D	Diâmetro do telescópio
δ	profundidade de uma imperfeição na superfície do espelho
F_{obj}	Função objetivo
$Y(r, \Theta)$	Cota do ponto (r, Θ)
Θ	Coordenada do sistema de coordenadas cilíndricas do espelho
Y_{ref}	Cota de uma parábola de referência
$del(r, \Theta)$	Diferença de cota entre pontos adjacentes
$B(u)$	Valor da curva de Bézier no ponto u
k	Número do ponto de controle
P_k	Cota do ponto de controle " k "
N	Número de pontos de controle menos 1
$B(u, v)$	Valor da curva de Bézier no ponto u, v
i	Número do ponto de controle em uma direção
j	Número do ponto de controle na direção perpendicular
$P_{i,j}$	Cota do ponto de controle " i, j "
N_i	Número de pontos de controle em uma direção menos 1
N_j	Número de pontos de controle na direção perpendicular menos 1

1- Introdução

Antes da invenção do telescópio, a humanidade vivia em um mundo envolvido pelo mistério. O céu coberto de estrelas e os estranhos movimentos dos planetas inspiravam e amedrontavam, e sem saber o que eram aquelas luzes errantes entre as estrelas fixas, as pessoas os tomavam por deuses e os temiam. A invenção do telescópio veio a mudar essa condição.

Em 1608, Hans Lippershey, um fabricante de lentes holandês, segurou duas lentes frente a frente e notou que as construções distantes pareciam mais próximas. Logo, ele percebeu que sua descoberta tinha aplicações importantes, e passou a produzir simples lunetas para seu governo, que as usava como instrumento de guerra.

Logo, a notícia de tão fantástico instrumento se espalhou, apesar dos esforços do governo Holandês de manter sua secreta a sua nova arma. Em 1609 a notícia chegou à Itália, e em junho daquele ano, Galileu Galilei construiu seu primeiro telescópio. Ele aumentava três vezes. Em agosto, ele havia construído um telescópio mais potente, que aumentava 8 vezes, e em novembro, com um telescópio que aumentava vinte vezes, ele transformou o mundo para sempre.



Fig. 1: Dois dos telescópios
construídos por Galileu

Naquele mês, Galileu voltou seu telescópio para o céu, e em poucos dias, destruiu a concepção vigente de Universo e do lugar da Terra. Ele descobriu que a lua não era etérea nem perfeita, muito menos uma deusa. A lua era um *Lugar!* Um lugar vasto, com vales, cordilheiras, planícies e estranhos buracos. E descobriu que Júpiter tampouco era um deus, mas uma esfera. Uma grande e distante esfera, seguida por um séquito de quatro pequenas luas. Eram os primeiros corpos que claramente não orbitavam a terra, a louvar o ápice da criação. Ao apontar seu telescópio para a Via-Láctea, pequenas manchas de nebulosidade se revelaram oceanos de estrelas, nunca vistas por olhos humanos. Nem mesmo o Sol pôde ocultar que era coberto de pequenas manchas, e girava.

Hoje, temos à nossa disposição uma incrível gama de instrumentos, e com eles foi possível estender as fronteiras do nosso universo para além do que supunha o mais desvairado sonho.

2- Classificação de telescópios

Existem basicamente três tipos de telescópios: Refratores, que utilizam lentes para trazer a luz para o foco, Refletores, que utilizam espelhos curvos e Catadióptricos, que utilizam ambos.

2.1- Telescópios refratores:

Os telescópios refratores, também conhecidos como lunetas, foram os primeiros a serem inventados. Eles são extremamente simples, consistindo em apenas um tubo com uma lente (ou um conjunto de lentes) responsável por levar a luz ao foco.



Fig.2: Telescópio Refrator

Os telescópio refratores, porém, apresentam alguns problemas graves. Um deles é que ao passar pelas lentes, a luz se divide como em um prisma, e cada cor é lavada ao foco em uma posição diferente. Como resultado, a imagem aparece indistinta, com halos coloridos em volta dos objetos. Para sanar esse problema, passou-se a utilizar conjuntos de lentes agrupadas em pares, sendo que cada uma das lentes de um par deve ter um índice de refração diferente. Esses telescópios apresentam menos aberração cromática, e são chamados de "Acromáticos". Ainda há, porém, aberração cromática

residual, e para eliminá-la foi criado o telescópio "Apocromático", que é muito semelhante ao anterior, mas utiliza vidros de composições muito diferentes em suas lentes. Esses telescópios possuem alta qualidade de imagem, mas a um grande custo financeiro.

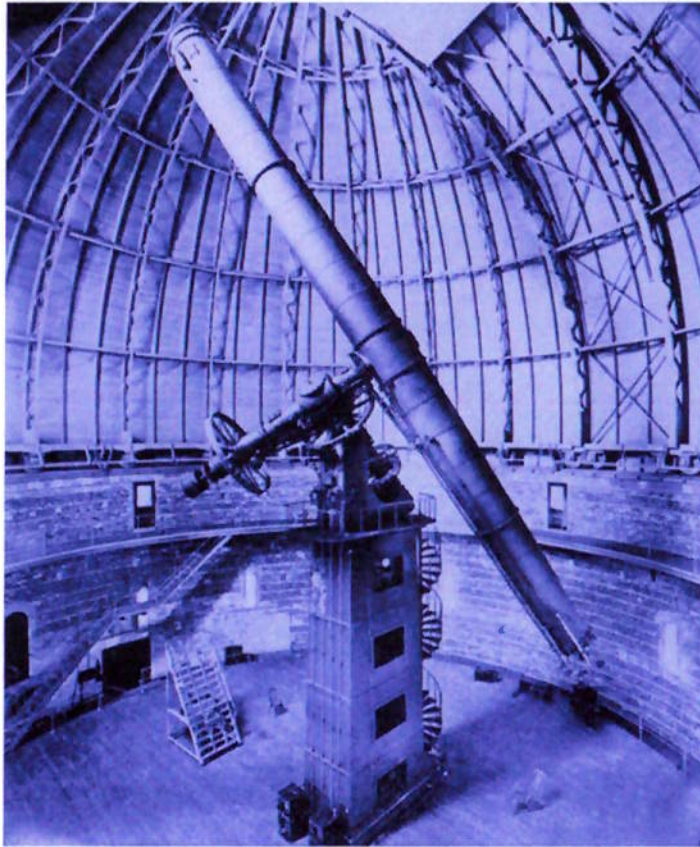


Fig.3: Telescópio Refrator de Yerkes, o maior do mundo

Outro problema com os telescópios refratores é o comprimento de seus tubos, o que limita o seu tamanho, pois torna-se inviável economicamente construir telescópios de grandes aberturas. O maior refrator já construído possui um metro de abertura, modesto para os padrões atuais, mas seu tubo possui mais de 19 metros. Ele foi construído em 1897, e devido ao custo muito menor dos telescópio refletores, continuará sendo a maior luneta do mundo para sempre.

2.2- Telescópios refletores

O telescópio refletor foi criado por James gregory, em 1663. Seu telescópio possuía dois espelhos côncavos. O espelho primário refletia a luz em direção ao secundário, que por sua vez a refletia através de um orifício na superfície do primário.

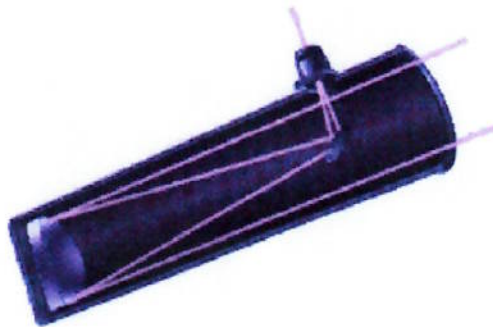


Fig4: Telescópio Refletor Newtoniano

Isaac Newton criou um telescópio mais simples, consistindo em um espelho primário parabólico que focaliza a luz em direção a um pequeno espelho secundário plano, montado inclinado de 45° em relação ao primário, a luz, assim, é direcionada para a lateral do tubo, onde se monta a ocular.



Fig.5: O Telescópio de Isaac Newton

Há centenas de variações de telescópios refletores, sendo que cada modelo diferente tenta minimizar uma ou mais das chamadas "aberrações", distorções da imagem obtida no plano focal do telescópio, assim, cada telescópio é indicado para diferentes tipos de observação.

Os modelos de telescópios refletores de maior sucesso são o telescópio Newtoniano, por sua enorme simplicidade e baixo custo, o telescópio Cassegrain, e o Richey-Chrétien. O Cassegrain é composto por um espelho primário côncavo parabólico com uma perfuração no centro, e um espelho secundário convexo hiperbólico. Ele possui um tubo mais curto do que o Newtoniano para uma mesma abertura, e uma qualidade de imagem superior. Porém, é mais complexo de ser produzido e portanto mais caro. O telescópio Richey-Chrétien é muito semelhante ao Cassegrain, mas seus dois espelhos são hiperbólicos. Ele apresenta a melhor qualidade de imagem de todos os tipos de telescópios, e pode ser produzido em qualquer abertura que se deseje. Como a superfície hiperbólica é de difícil produção, ele apresenta um custo ainda superior ao Cassegrain. Apesar disso, a grande maioria dos telescópios profissionais do mundo (inclusive o telescópio espacial Hubble) são Richey-Chrétien.

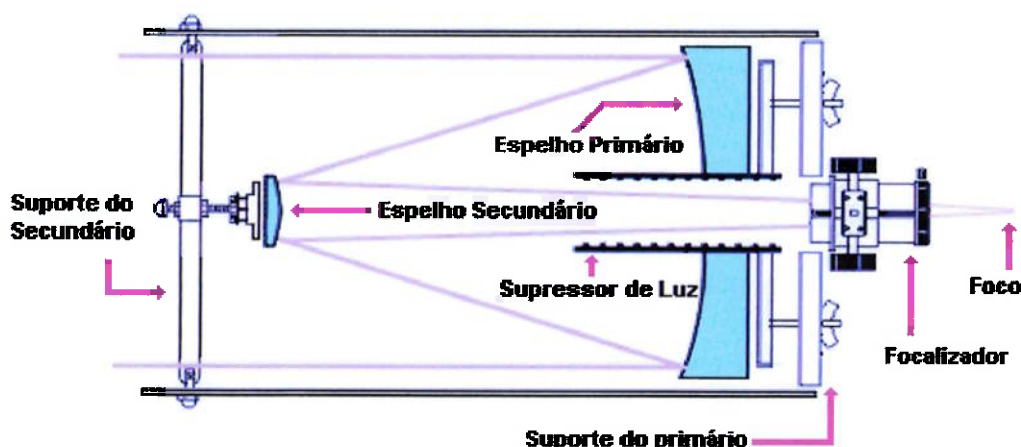


Fig. 6: Esquema de telescópio Cassegrain ou Richey-Chrétien

Os telescópios refletor são mais simples e baratos do que os refratores, e portanto, os maiores telescópio do mundo são, hoje, refletor. Telescópios com espelhos de mais de oito metros de diâmetro já são relativamente comuns, e já existem telescópios de espelhos segmentados que somam mais de dez metros de abertura.

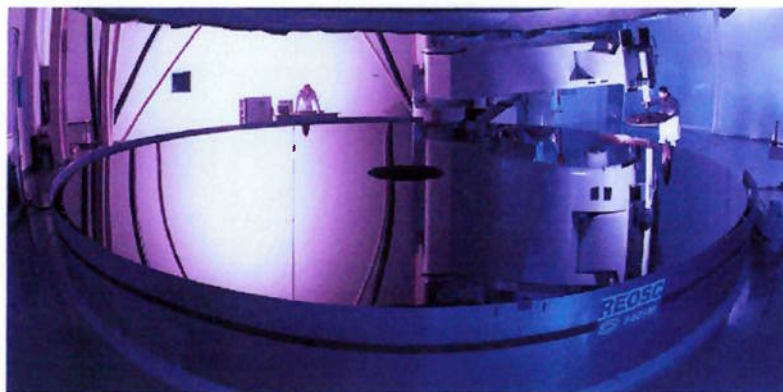


Fig.7: Espelho do telescópio VLT, no Chile, com 8,2m

O maior telescópio do mundo ainda está em construção, no Chile, e pertence à Agência Espacial Europeia. Ele é composto de 4 telescópios de 8,2 m de abertura, que combinam a luz por eles coletada em um único detector.



Fig. 8: Telescópio VLT. Cada um dos domos é mais alto do que um prédio de dez andares

2.3- Telescópios Catadióptricos

Telescópios catadióptricos nada mais são do que telescópios mistos, que utilizam combinações de lentes e espelhos para a formação da imagem. São telescópios de tamanho limitado, e pouco usados para a astronomia profissional. No mercado amador, porém, esses telescópios são extremamente populares, pois são relativamente baratos, possuem aberturas razoáveis e são muito compactos, podendo ser facilmente transportados.

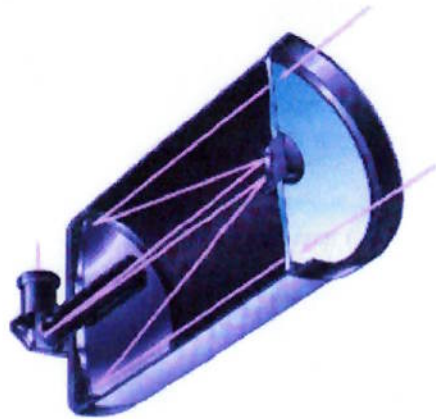


Fig.9: Telescópio Catadióptrico

3- Os espelhos tensionados: Breve histórico

Para que um telescópio produza imagens de qualidade, independentemente de que tipo de instrumento ele seja, suas superfícies óticas devem ser, para todos os efeitos, perfeitas. Um telescópio Newtoniano, por exemplo, precisa que seu espelho possua uma superfície que não se afaste de um parabolóide de revolução ideal mais do que um décimo do comprimento de onda da luz. Isso significa uma superfície com uma precisão de $5 \cdot 10^{-5}$ mm! Uma superfície parabólica ou hiperbólica com tal precisão é extremamente difícil de se obter, e portanto, cara. Curiosamente, a obtenção de uma superfície esférica com essa precisão é imensamente mais simples. Isso se deve ao fato de que o processo de polimento de um espelho naturalmente produz uma esfera, e é a partir do momento em que uma esfera é obtida que se alteram os padrões de polimento para que ela lentamente se torne um parabolóide, e a partir daí, um hiperbolóide. A superfície esférica é ainda muito mais simples de ser testada, e qualquer desvio de esfericidade é prontamente detectável, o que não ocorre com outras superfícies.

Um modo extremamente interessante de se obter superfície parabólicas foi proposto, em 1992, por William Kelley. Seu método consiste em produzir um espelho esférico e tensioná-lo, transformando-o em um parabolóide. O espelho deveria ser suportado por um anel localizado na periferia de sua face inferior e, através de um pino colado em seu centro, tensionado. Este método é capaz de produzir superfícies razoáveis, desde que o espelho seja pequeno e tenha distância focal grande, como espelhos de no máximo 8 polegadas, com razão "f" (a relação da distância focal e da abertura) maior do que 6.

Apesar de que os resultados originais de Kelley deixem a desejar, seu método originou uma resposta entusiasmada da comunidade astronômica, e em novembro de 2000, Alan Adler publicou um artigo onde descrevia um novo método para flexionar espelhos de forma a obter parábolas melhores. Ao invés de "puxar" o espelho em um ponto central, Adler utiliza um largo anel de borracha que distribui a tensão igualmente por quase toda a superfície traseira

do espelho. Este método inovador é capaz de produzir parábolas de altíssima qualidade, mesmo em espelhos maiores e de menor distância focal.

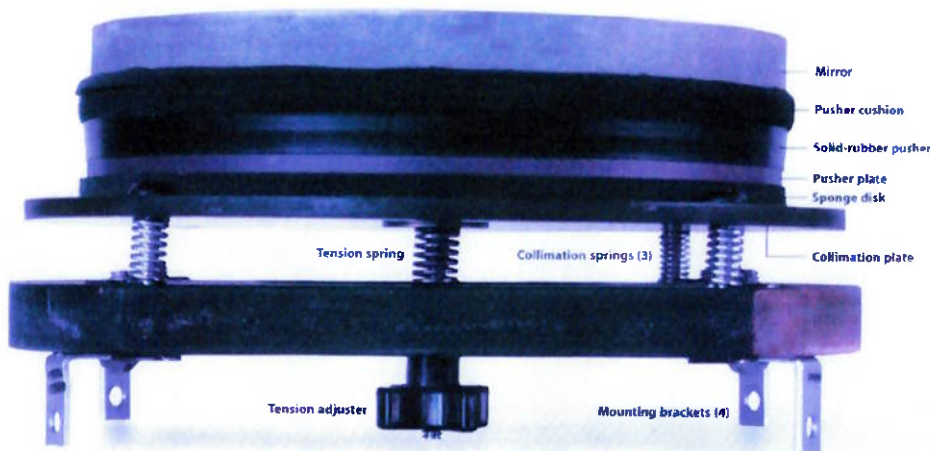


Fig.10: Espelho esférico montado em uma célula fletora

O método de Adler, conhecido como MicroFlex, apresenta, porém, alguns problemas: A menor imperfeição na confecção do dispositivo de flexão do espelho provoca graves aberrações na imagem, em especial, astigmatismo. Além disso, a fluência da borracha lentamente devolve ao espelho a sua forma esférica, e constantes ajustes da tensão são necessários. O fato de toda a face posterior do espelho ser coberta de borracha também ocasiona problemas: Com apenas metade de sua área exposta ao ar, o espelho leva mais tempo para se resfriar durante a noite. Diferenças de temperatura de apenas um grau entre o espelho e o ar criam pequenas células de convecção sobre sua superfície que arruinam a imagem.

O uso de vácuo para flexionar um espelho foi proposto, mas nenhum protótipo foi ainda construído. Tal método tem potencial para criar espelhos excelentes, devido à perfeita distribuição de tensões proporcionada pelo vácuo. Há, porém, uma série de complicações de ordem técnica já previstas, como por exemplo o cuidadoso controle da pressão, para evitar que alterações na pressão atmosférica causem distorções no espelho. A complexidade de um sistema de tensão por vácuo torna esse método pouco atrativo.

Alan Adler, em seu artigo, também descreve um novo método criado por Kelley: Como em seu método original, o espelho é apoiado em sua periferia e

tensionado pelo centro. Agora, porém, a face posterior do espelho é otimizada, de forma que a sua deformação crie uma parábola de altíssima precisão em sua face anterior. O primeiro protótipo deste novo método foi concluído em agosto de 2001, e os teste revelaram uma superfície essencialmente perfeita, com menores do que $1/40$ do comprimento de onda da luz, ou 1.10^{-6} mm!

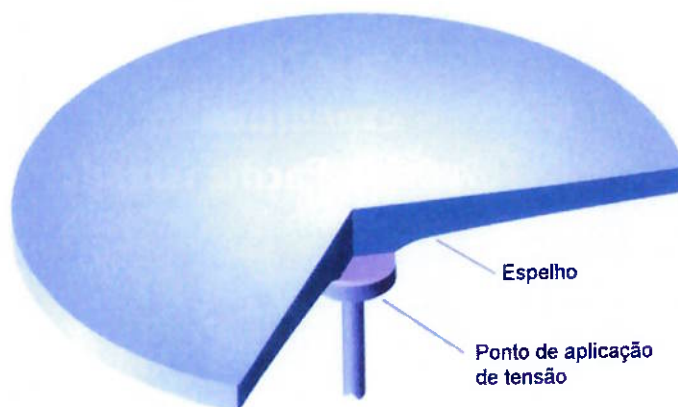


Fig.11: Diagrama do espelho de Kelley

O método de Kelley ainda resolve dois dos problemas que o método microflex possuía. Como não há mais borracha recobrindo a face de trás do espelho, os problemas térmicos ficam muito diminuídos, e a forma do espelho é mantida pois não há mais problemas de fluência. Porém, se o anel que suporta o espelho não for perfeitamente circular e não estiver muito bem posicionado, a imagem apresentará aberrações inaceitáveis.

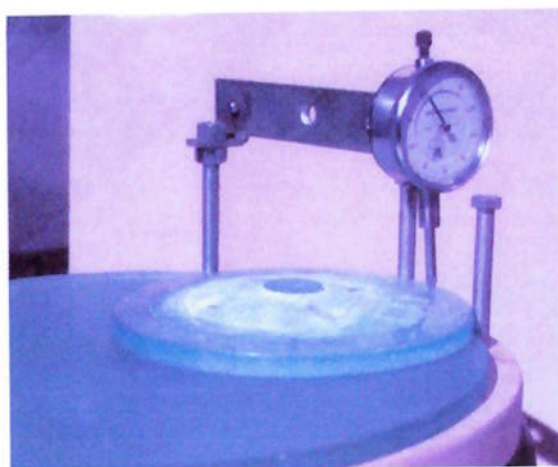


Fig. 12: Protótipo do espelho de Kelley

4- Proposta para este trabalho

Para tentar eliminar os problemas inerentes ao espelho de Kelley, proponho a criação de um espelho que, como o espelho de Kelley, seja tensionado em seu centro, mas que seja suportado por apenas 3 pontos em sua periferia, e não por um anel. A superfície traseira deste espelho seria radicalmente diferente daquela obtida por Kelley, e consideravelmente mais complexa, pois não será uma curva de revolução. Tal superfície, porém, poderá ser gerada por qualquer máquina CNC, sem grandes dificuldades.

O espelho será modelado por elementos finitos (FEM), sendo que o modelo será construído parametricamente. Dessa forma, um algoritmo de otimização, associado ao programa de FEM, irá alterar os parâmetros que descrevem a superfície do espelho, de forma a fazer com que a superfície frontal do espelho se torne o mais próximo possível de uma superfície esférica desejada.

Pode-se criar espelhos, com essa técnica, com os mais variados tipos de superfície, desde que se modifique a sua face traseira. Assim, pode-se fazer com que o espelho esférico se deforme para se tornar um parabolóide, um hiperbolóide, esferas oblatas ou prolatas (que nada mais são do que elipsóides de revolução em torno de cada um de seus eixos) ou até mesmo superfícies toroidais. Estas últimas, apesar de menos comuns, são utilizadas em alguns tipos de instrumentos de uso mais específico.

5- Aspectos econômicos

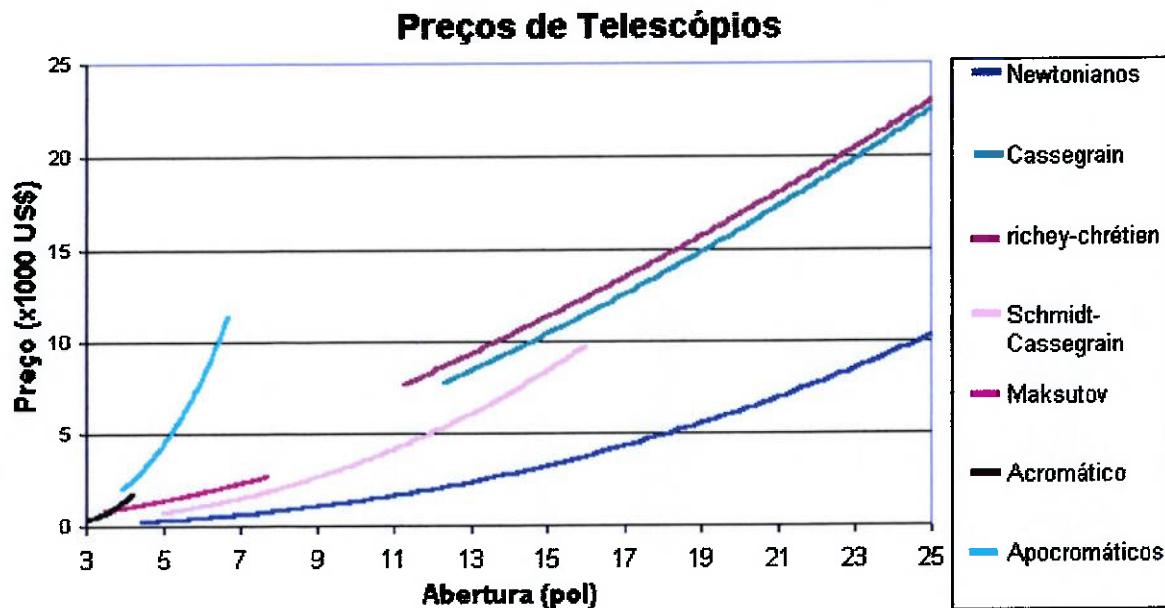


Gráfico 1: Preço de telescópios versus abertura

O gráfico acima mostra a relação entre custo e abertura para diferentes tipos de telescópios disponíveis para o astrônomo amador. O gráfico foi levantado a partir dos preços de 23 fabricantes diferentes. Pode-se ver que os telescópios refratores estão limitados a tamanhos pequenos, devido a sua rápida elevação de preços conforme a abertura aumenta. Pode-se notar ainda que os telescópios Newtonianos são os que apresentam a melhor relação custo-benefício, mas estão relativamente limitados pelo tamanho de seus tubos (os maiores telescópios Newtonianos plotados no gráfico têm tubos de mais de três metros de comprimento). Para se obter telescópios maiores, deve-se utilizar Cassegrains ou Richey-Chrétiens. O custo destes telescópios é muito maior do que o de um Newtoniano de mesma abertura, devido à dificuldade de obtenção de suas superfícies mais complexas. Com os métodos de espelhos tensionados, pode vir a ser possível a obtenção de telescópios Cassegrain e Richey-Chrétien por um custo apenas um pouco maior do que o dos Newtonianos.

6. Definição das características desejadas do espelho

A distância focal de um espelho esférico não é constante, e é isso o que impossibilita o seu uso em telescópios. A expressão abaixo fornece a distância em que um raio de luz cruza o eixo óptico de uma superfície de revolução qualquer após ser refletido pela superfície a uma distância r de seu eixo:

$$Y = r \cdot \text{tg}[90 - 2\text{arctg}S'(r)] + S(r)$$

Onde $S(r)$ é a equação da curva que ao ser rotacionada gera a superfície em questão.

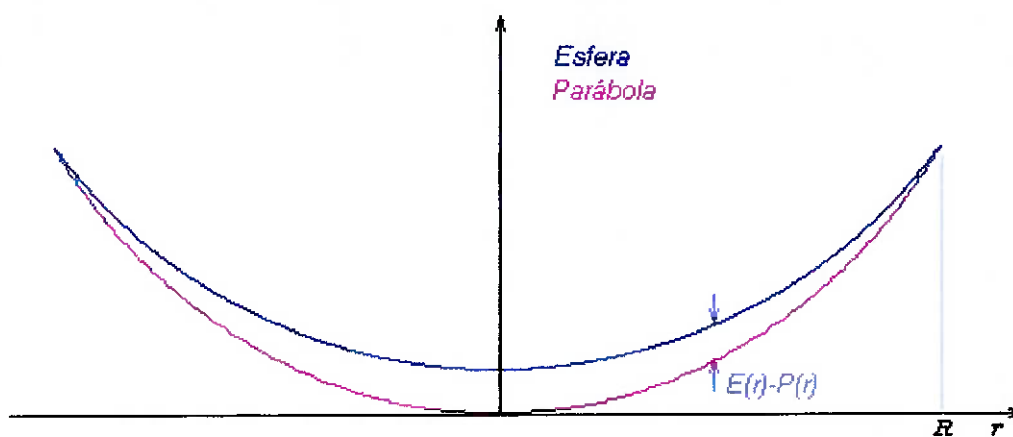


Fig. 13. Comparação entre a esfera e a parábola

O gráfico abaixo mostra a variação da distância focal de uma esfera em função da zona do espelho (o anel de raio igual a r) atingida pelo raio de luz.

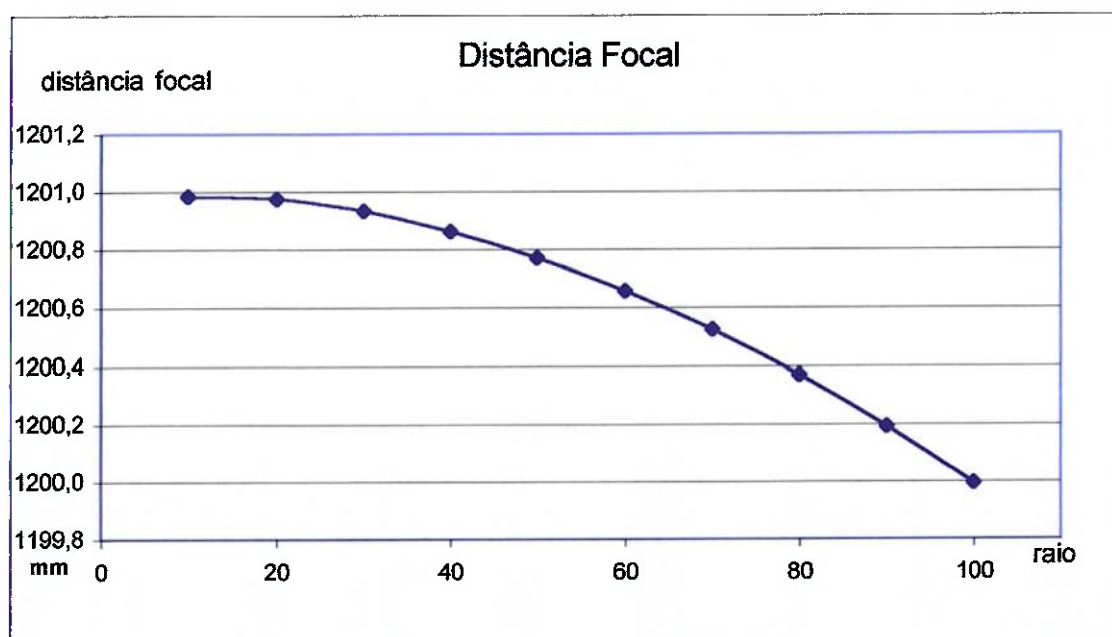


Gráfico 2. Aberração esférica

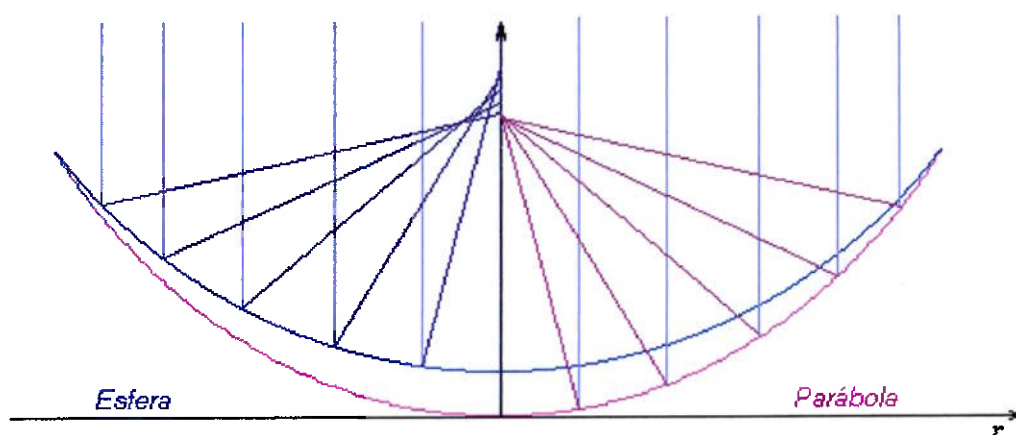


Fig. 14. Aberração esférica

A esfera e a parábola diferem muito pouco: A diferença de uma para outra é de menos de um milésimo de milímetro para um espelho de 20 cm de diâmetro e razão f igual a 6 (ou seja, distância focal/diâmetro = 6). Por menor que pareça, tal diferença causa enormes alterações na qualidade da imagem, a tal ponto que um espelho esférico de mesmo diâmetro e razão f é praticamente inútil. O gráfico abaixo fornece a diferença entre a esfera e a parábola para este espelho. Estes valores são exatamente o deslocamento que deve ser obtido ao se flexionar o espelho.

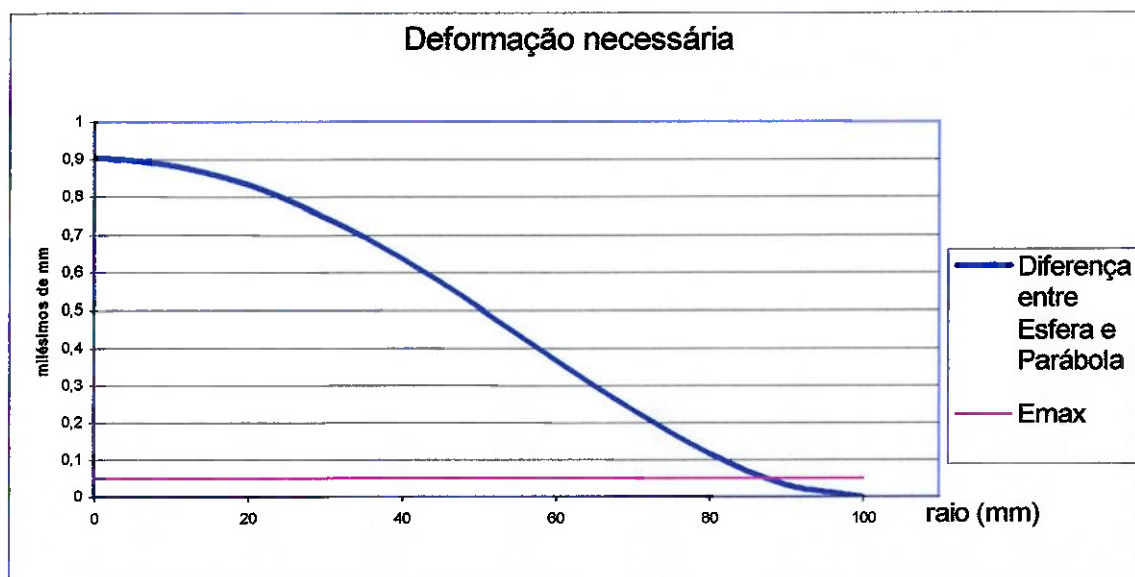


Gráfico 3. Deformação necessária

A linha vermelha no gráfico acima, Emax, representa o erro máximo aceitável na flexão do espelho. A precisão necessária para a formação de uma imagem adequada será discutida na próxima sessão.

Para que o espelho forme uma imagem de uma estrela sobre seu eixo óptico, é necessário que sua superfície seja um parabolóide de revolução¹. A parábola que ao ser rotacionada sobre seu eixo produz o parabolóide desejado é definida pela seguinte expressão:

$$P = \frac{r^2}{4f_p}$$

Onde f_p é a distância focal da parábola e r é a distância do ponto em questão ao centro do espelho. Desejamos, porém, obter este parabolóide através da flexão de uma esfera, e é necessário determinar as características dessa esfera. A esfera deverá ser suportada por sua extremidade, portanto a flexão em sua periferia será nula, ao passo que será muito pronunciada em seu centro, onde é aplicada a força. Assim, uma esfera que tangencia a

¹ No jargão astronômico, tal espelho é chamado apenas de *Parabólico*.

parábola em sua extremidade é ideal. Tal esfera possui as seguintes características:

$$E(R) = P(R)$$

e

$$E'(R) = P'(R)$$

Ou seja, tanto a esfera quanto a parábola possuem mesmos valores e derivadas na extremidade do espelho ($r = R$)., como se pode notar na figura 13. Podemos, assim, tomar a equação de um círculo de centro (0,0) e raio R_c

$$E(r)^2 + r^2 = R_c^2$$

e reescrevê-la da forma:

$$E(r) = -\left(R_c^2 - r^2\right)^{1/2} + \Delta h \quad -R_c < r < R_c$$

Note que o centro não é mais (0,0), mas fazendo Δh adequado agora o círculo tangencia a parábola no ponto $r=R$, e apresenta concavidade para cima. Tomemos agora $E(r)=P(r)$ para $r=R$

$$-\left(R_c^2 - R^2\right)^{1/2} + \Delta h = \frac{R^2}{4f_p}$$

E por fim, tomemos $E'(r)=P'(r)$ para $r=R$

$$\Delta h = \frac{R^2}{4f_p} + \left(R_c^2 - R^2\right)^{1/2}$$

$$\frac{R}{\left(R_c^2 - R^2\right)^{1/2}} = \frac{R}{2f_p}$$

$$R_c = \left(4f_p^2 + R^2\right)^{1/2}$$

Temos, assim, que a equação da esfera desejada, que tangencia a parábola no ponto $r=R$, é:

$$E(r) = -\left(R_c^2 - r^2\right)^{1/2} + \frac{R^2}{4f_p} + \left(R_c^2 - R^2\right)^{1/2}$$

sendo que o raio de curvatura da esfera é:

$$R_c = \sqrt{4f_p^2 + R^2}$$

7. Sobre a precisão necessária

A qualidade da imagem de um telescópio está limitada, em condições ideais, à natureza ondulatória da luz. Ao se aumentar muito a imagem de uma estrela, percebe-se que ela não é um ponto matemático, como se esperaria que fosse a imagem formada por raios de luz paralelos, mas um pequeno disco, rodeado por vários anéis que decrescem em luminosidade conforme sua distância do centro aumenta. Esse padrão, conhecido como padrão de difração, determina a resolução máxima de um telescópio, pois nenhum detalhe menor que o disco central (conhecido por disco de Airy, em homenagem ao seu descobridor) será visível.

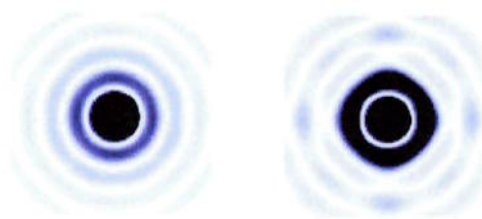


Fig. 15. Padrões de difração. À esquerda: Telescópio não -obstruído. À direita: Telescópio com obstrução por espelho secundário e seu suporte. Note que os efeitos da difração são muito mais pronunciados nesse caso.

O diâmetro do disco de Airy, por sua vez, depende apenas da abertura do telescópio e do comprimento de onda sendo observado. Para um determinado comprimento de onda, o aumento da abertura do telescópio reduz o diâmetro do disco de Airy, e a resolução do aparelho aumenta, como seria de se esperar. A expressão abaixo fornece o tamanho angular do disco de Airy (ρ).

$$\rho = 1,22 \frac{\lambda}{D}$$

Onde D é o diâmetro do telescópio, em metros, e λ é o comprimento de onda da luz sendo observada.

A tolerância da superfície de um espelho para telescópios é definida pelo diâmetro do disco de Airy, pois após certo ponto, qualquer aumento na qualidade da superfície deixa de ser visível, oculto pelos "erros" criados pela natureza ondulatória da luz. Em pesquisas realizadas por Lorde Rayleigh, ele verificou que se a frente de onda que forma a imagem se alterasse em até um quarto do comprimento de onda, a imagem final era alterada, mas ainda bastante próxima do ideal. Para alterações maiores na frente de onda, a imagem rapidamente se degradava além do aceitável. Esse limite proposto por Rayleigh apenas especifica o erro máximo que a frente de onda pode ter, mas não diz nada quando ao máximo erro tolerável na superfície do espelho propriamente dito.

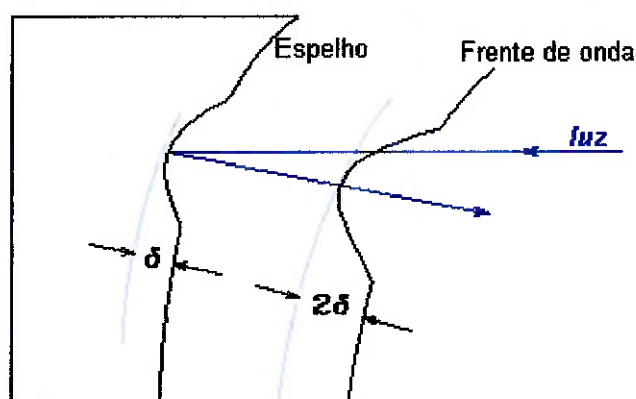


Fig. 16. Frente de onda

Se a superfície de um espelho tiver um defeito de profundidade δ , a luz que incidir sobre este defeito deverá percorrer uma distância 2δ a mais do que a luz que incide sobre a parte não defeituosa. Assim, a frente de onda apresentará um defeito de magnitude 2δ . Como o máximo erro aceitável é de um quarto do comprimento de onda, conclui-se que o erro máximo tolerável na superfície do espelho é de um oitavo de comprimento de onda.

Há, porém, outros aspectos a considerar: Danjon (1933)² observou que deve-se levar em conta a imperfeição total da frente de onda, ou seja, é preciso considerar também erros causados pela atmosfera, o que reduz ainda

² Danjon, A. Réunions Institut d'Optique, 2e réunion. Editions de la revue d'optique théorique et instrumentale, Paris, 1934.

mais a tolerância a erros na superfície do espelho. Couder (1944)³ mostrou que a forma e o número de defeitos também podem influenciar fortemente a qualidade da imagem. Se a superfície do espelho possuir muitos defeitos de pouca profundidade, mas de grande "active", eles poderão espalhar parte considerável da luz para regiões fora do disco de Airy, degradando enormemente a imagem.

Francon (1947)⁴ notou, ainda, que o critério de Rayleigh se aplica muito bem à imagens brilhantes e pontuais, como estrelas, mas ao se observar imagens de baixo contraste, como detalhes na superfície dos planetas, a tolerância deve ser muito mais rígida. Imperfeições de $1/32$ do comprimento de onda na superfície do espelho já produzem deterioração perceptível na imagem.

Obviamente, uma superfície de tal precisão é extremamente difícil de ser conseguida, e mesmo que se fosse obtida, apenas em poucos locais do globo se pode encontrar condições de estabilidade atmosférica de qualidade correspondente. Em geral, a maioria dos fabricantes adota como tolerância, um valor de 0,1 do comprimento de onda.

Para transformar esse valor em unidades métricas, é preciso escolher um comprimento de onda de referência. Em geral, se escolhe um tom de verde-amarelado, pois tal cor corresponde ao máximo de sensibilidade do olho humano. Para este trabalho, será adotado o valor de 5007 angstroms, pois tal valor, além de se situar próximo ao centro da faixa de maior sensibilidade do olho, corresponde à emissão do oxigênio triplamente ionizado, a emissão de nebulosas mais comum que pode ser observada visualmente.

Assim, temos que a tolerância usualmente aceita pela indústria é de apenas 0,05 milésimos de milímetro, ou 50nm.

³ A. Couder, *Cashier Phys.*, N° 26, dezembro de 1944

⁴ Francon, *Rev. Opt.*, 26, N° 10, 1947

8. O teste de foucault

A medição de variações tão pequenas na superfície de um espelho pode parecer um tarefa de grande dificuldade, mas existe um método extremamente simples e preciso de realizá-la: O teste de Foucault.

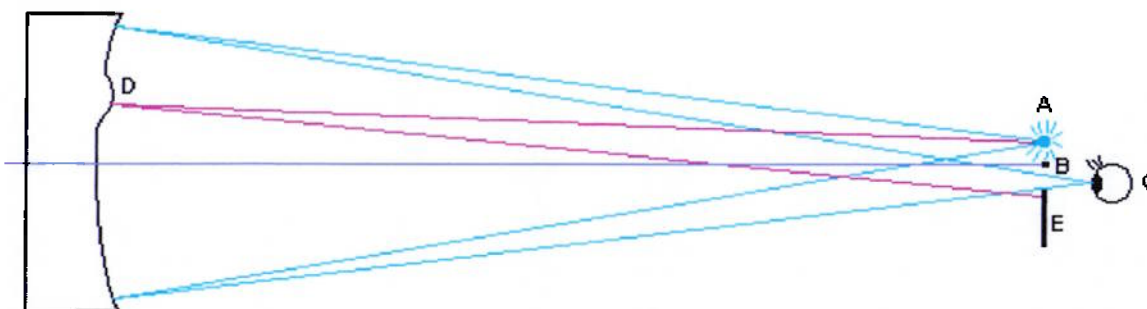


Fig. 17. Esquema de montagem de um Teste de Foucault

Neste teste, a luz proveniente de uma fonte pontual (A), situada imediatamente ao lado do centro de curvatura do espelho (B), é refletida pelo espelho (raios verdes) e atinge o olho do observador (ou uma câmera fotográfica, em C). A luz que atinge uma imperfeição da superfície do espelho (D), tem seu trajeto desviado (raios vermelhos) e é bloqueada pelo anteparo E, não atingindo o observador. Dessa forma, todas as imperfeições na superfície do espelho são facilmente reveladas.

O teste de Foucault é um teste ideal para espelhos esféricos, uma vez que como a fonte luminosa é colocada em seu centro de curvatura, a imagem que o observador percebe é um disco igualmente iluminado. Qualquer imperfeição na superfície pode ser imediatamente identificada. Se um espelho parabólico for testado nas mesmas condições, sua imagem não será de interpretação tão simples, pois além das imperfeições na superfície, será também revelada a diferença de curvatura da esfera e da parábola, e é preciso saber separar os dois efeitos. Conseqüentemente, é muito mais simples realizar o teste de um espelho esférico.

A simplicidade de se testar um espelho esférico é uma das grandes vantagens do método de espelhos tensionados, pois toda a fase de testes e

correções do espelho é feita com ele em sua forma esférica, e ele só recebe a tensão que o tornará parabólico após esta etapa estar terminada. Dessa forma, é muito mais simples identificar os defeitos em sua superfície, e corrigi-los. Com isso é possível obter espelhos de qualidade não só equivalente, mais muito superior do que os espelhos produzidos por métodos convencionais, e a um custo menor.

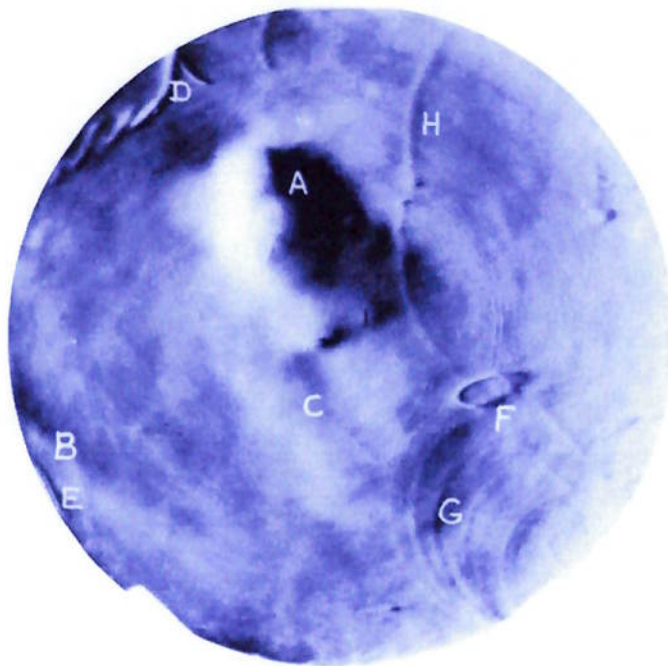


Fig. 18. Exemplo de teste de Foucault

A ilustração acima mostra uma foto de um teste de Foucault de um espelho de 400 mm de diâmetro. Se pode notar facilmente uma série de defeitos, marcados com letras, que são totalmente óbvios no teste (A mostra uma grande elevação causada por não homogeneidade na composição do vidro. B a H mostram diversas estrias). Nenhum desses defeitos, porém, apresenta uma diferença de altura em relação à superfície ideal maior do que 50 nm, o que revela a sensibilidade do teste.

9. Aproveitamento da simetria

O número de variáveis pode ser bastante diminuído, e o problema simplificado, se for aproveitada a simetria do espelho. O espelho não apresenta uma simetria axisimétrica, pois é suportado em três pontos, mas apresenta uma simetria tri-radial, em que cada "gomo" do espelho pode ainda ser dividido em dois, pois cada uma das metade é exatamente igual à outra, apenas espelhada.

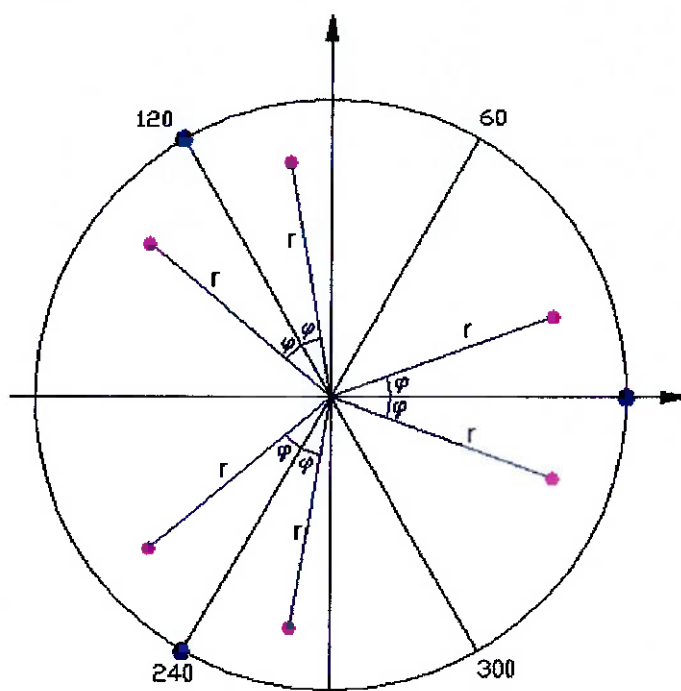


Fig. 19. Simetria do modelo

Na ilustração acima, os pontos azuis representam as posições dos suportes, enquanto que os pontos vermelhos representam pontos da superfície traseira do espelho que, por simetria, possuem a mesma cota. Estes seis pontos podem, assim, ser descritos pela mesma variável.

Aproveitando a simetria do modelo, como mostrado acima, é possível reduzir o número de variáveis em um sexto do original. Com isso, o problema é simplificado, o tempo de processamento é reduzido e as chances de se atingir um mínimo local ao invés de um global são bastante diminuídas.

10. Definição da função objetivo e restrições

O programa de otimização busca sempre, ao alterar o valor das variáveis, minimizar a função objetivo, que é uma função das variáveis de projeto. Para este trabalho, a função objetivo é uma avaliação da conformidade da superfície deformada do espelho se comparada com uma parábola ideal. Essa avaliação pode ser feita de diversas formas, mas nem todas são adequadas para o programa de otimização, assim, é preferível enunciar a função como mostrado abaixo:

$$F_{obj} = \sum_{i=1}^n [uy_i - (E(r) - P(r))]^2$$

Onde $Y(r, \Theta)$ é a cota do ponto de coordenadas r e Θ , e Y_{ref} é a cota de uma parábola de referência.

O tipo de otimização escolhida foi o método de primeira ordem, que utiliza informações sobre a derivada da função objetivo para decidir como alterar as variáveis. Esse método é bastante preciso, mas exige grande capacidade de processamento.

Para evitar que a solução do problema recaia sobre um mínimo local, antes de aplicar o método de primeira ordem serão gerados diversas soluções aleatoriamente, e a partir delas será buscada a solução otimizada.

Nem todas as soluções, porém, são viáveis. Há algumas restrições que devem ser obedecidas, como por exemplo, a espessura do espelho. Caso o espelho seja fino demais, surgem dificuldades para lixar e polir a superfície, pois o vidro será deformado durante o processo. Caso o espelho seja muito espesso, porém, ele deixa de ser prático, além de que ele levará um tempo muito grande para se resfriar durante a noite, o que impõe sérios problemas de convecção sobre a sua superfície, degradando a imagem. Além de restrições na espessura máxima e mínima do espelho, é adequado restringir também a variação entre pontos adjacentes, pois variações bruscas na superfície traseira do espelho não produzirão resultados adequados.

Dessa forma, o problema de otimização pode ser enunciado da seguinte forma:

Minimizar:

$$F_{obj} = \sum_{i=1}^n [u_{y_i} - (E(r) - P(r))]^2$$

Onde u_{y_i} é o deslocamento do ponto i na direção do eixo y ,

Tal que:

A espessura do espelho esteja entre 10 e 50 mm, e:

$$0 \leq del_i \leq 10$$

Onde del_i é a diferença de cota entre pontos adjacentes

11. Resultados

Como uma primeira aproximação ao problema, foi utilizado não o modelo suportado em três pontos, mas um modelo suportado em toda a sua periferia, pois este é consideravelmente mais simples, uma vez que é axisimétrico. A solução para este problema já existe, e foi proposta por Alan Adler em 2000. O propósito de reproduzir este resultado é apenas sondar as dificuldades do problema em questão e verificar a validade dos resultados obtidos.

11.1. Escolha de variáveis

A escolha adequada das variáveis do problema se mostrou decisiva para o bom condicionamento do problema e a qualidade das soluções obtidas. O problema que se coloca é qual deve ser a melhor maneira de se definir uma curva de complexidade suficiente utilizando-se de um número relativamente pequeno de variáveis. A curva deve ser capaz de assumir a maior gama possível de geometrias para possibilitar que o programa de otimização possa explorar as mais diversas soluções possíveis. É desejado, por outro lado, que o número de variáveis não seja muito elevado, pois do contrário o gasto computacional será maior.

11.1.1 Diferenças de cotas como variáveis

A utilização das diferenças de cotas entre pontos adjacentes como variáveis de projeto foi a idéia original para a definição de variáveis. Este tipo de variável é recomendada pelo software Ansys como a mais adequada para problemas deste tipo.

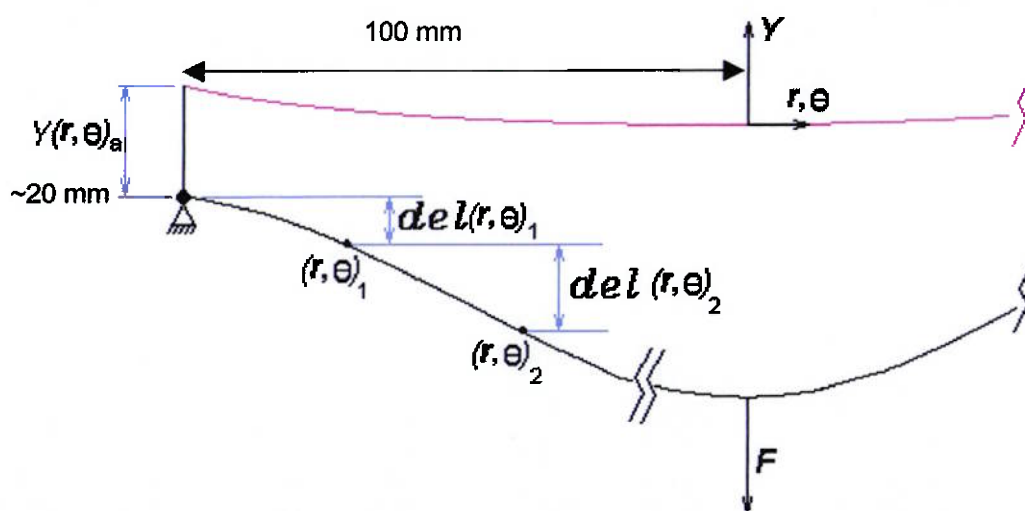


Figura 20. Diferença de cotas como variáveis

As cotas de cada ponto, assim, são definidas não só por uma variável associada ao ponto em questão, mas pelas variáveis associadas a todos os pontos “anteriores” a ele. A cota do ponto 4, por exemplo, é dada pela soma da espessura na periferia do espelho somadas às diferenças de cota entre a periferia (ponto 1) e o ponto 2, entre este e o ponto 3 e finalmente entre o ponto 3 e o ponto 4. Quando definidas desta forma, as variáveis “amarram” a geometria obtida, pois a mudança do valor de apenas uma variável altera a geometria de toda a curva.

Para este problema, utilizou-se 10 variáveis para definir a geometria e uma para a força. Este número ocasionou problemas de pequena discretização da malha, associada ao problema da criação de uma curva angulosa. Assim, os resultados obtidos por este método foram bastante inadequados.

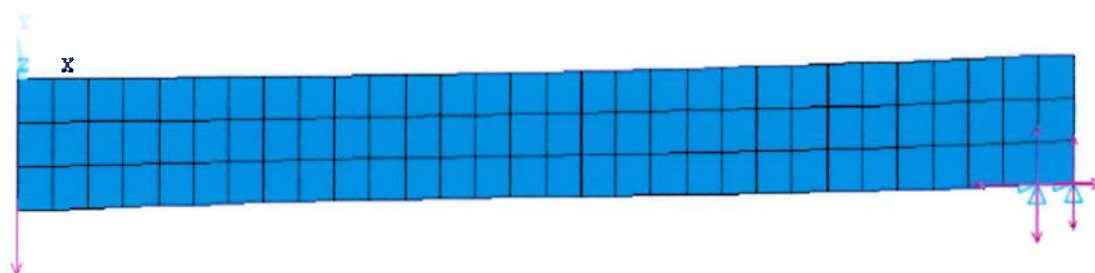


Figura 21. Resultados obtidos

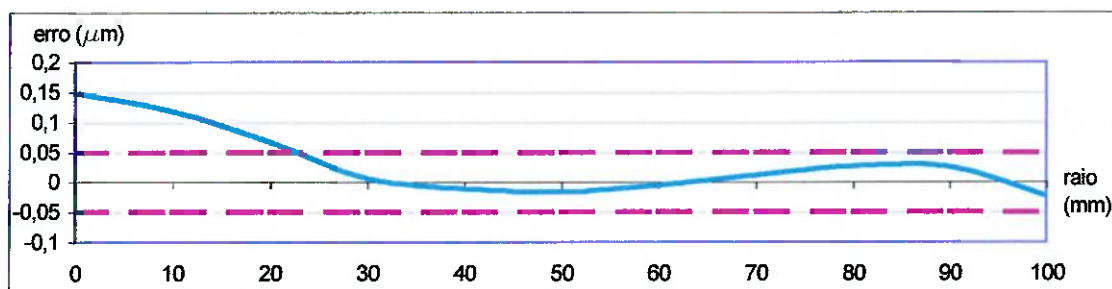


Gráfico 4. Planilhas de resultados

Acima, estão mostrados os resultados da otimização utilizando-se o conjunto de variáveis descrito. No gráfico superior, o eixo das abscissas representa uma parábola Euclidiana. A linha verde cheia representa a superfície superior do espelho quando deformado. As linhas vermelhas tracejadas, paralelas, representam, o máximo erro tolerado ($\lambda/10$). Assim, pode-se comparar a superfície obtida após a aplicação da força com uma parábola ideal, levando em consideração a tolerância aceitável. Pode-se verificar que o resultado gerado pela otimização não é aceitável.

O problema de otimização se revelou repleto de mínimos locais, o que torna a otimização particularmente difícil. A partir de um conjunto de valores iniciais para cada uma das variáveis, o programa freqüentemente caía em um mínimo local após pequenas (e em geral inexpressivas) alterações nestes valores. Assim, o problema se mostrou muito dependente dos valores definidos inicialmente, e, quando muito, a solução podia ser encarada apenas como um refinamento destes valores, sem que houvesse efetivamente a geração de uma geometria nova.

11.1.2 Cotas como variáveis

Apesar de não recomendado pelo Ansys, as variáveis foram, em seguida, definidas como a cota de cada ponto da curva. Este método dá à curva a maior liberdade possível, e não restringe nenhuma geometria. A curva, porém está sujeita a adquirir uma geometria irregular e muito angulosa (o que de fato aconteceu)

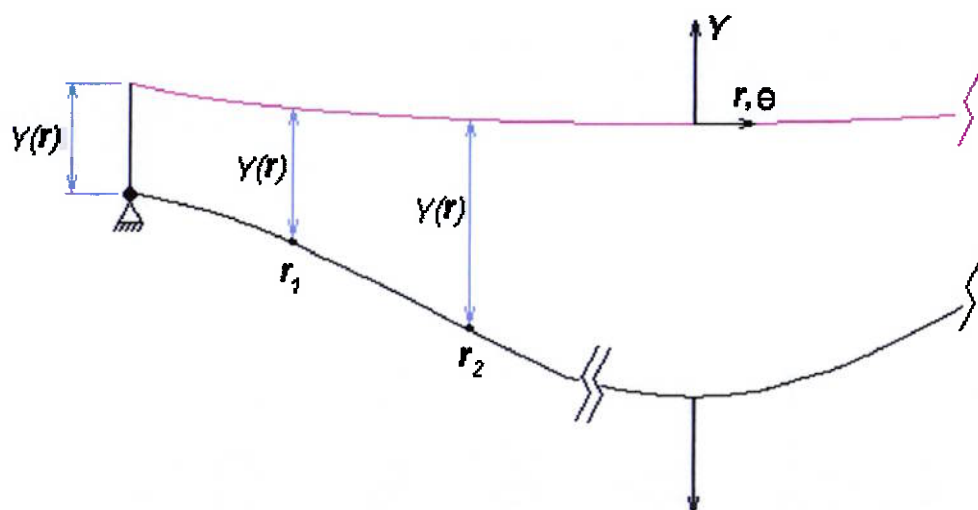


Figura 22. Cotas como variáveis

O número de variáveis foi mantido, de forma que o problema de discretização da malha foi mantida, bem como o problema da criação de uma curva angulosa, que foi até aumentado. Quando se utiliza diferenças de cotas como variáveis, a curva é sempre crescente, pois a diferença deve sempre ser maior do que zero, por uma limitação do software Ansys. Isto não ocorre se as variáveis forem as próprias cotas, caso em que a curva pode tanto crescer como decrescer.

ANSYS

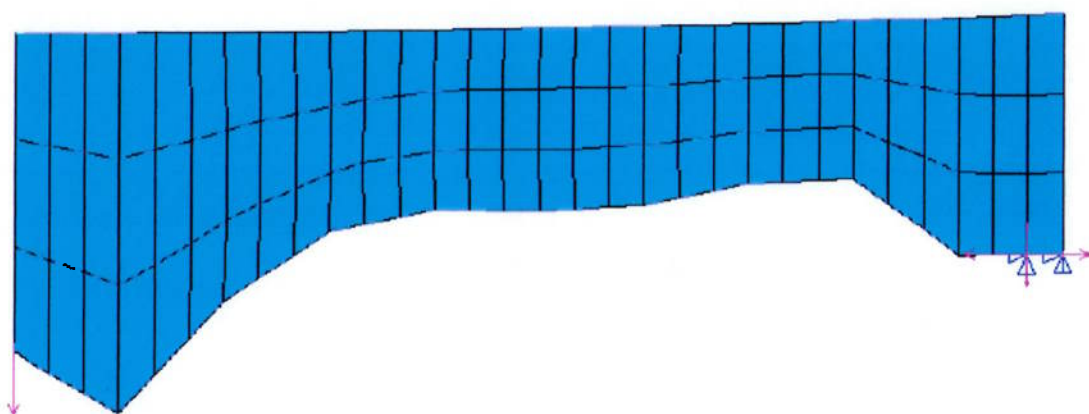


Figura 23. Resultados obtidos

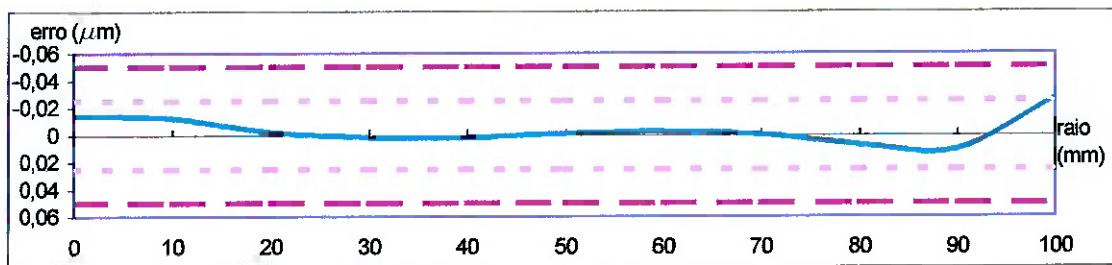


Gráfico 5. Planilha de resultados

Os resultados obtidos com este método foram muito superiores aos alcançados anteriormente, como se pode ver acima. Foi obtida uma parábola cuja diferença de uma parábola de referência foi de apenas 1/18 do comprimento de onda da luz, um resultado respeitável. As linhas horizontais laranjas pontilhadas representam um erro de 1/20 do comprimento de onda da luz. Pode-se notar que em grande parte da superfície o erro é praticamente desprezível. As imagens que poderiam ser produzidas por esse espelho, se ele fosse construído, seriam excepcionais.

Ainda assim, continuaram os problemas de mínimos locais, de forma que a solução da otimização continua extremamente dependente dos valores escolhidos inicialmente. Além disso, não foi possível obter a solução acima de modo imediato, uma vez que notou-se que ao incluir a força como variável de projeto, a otimização encontra muita dificuldade para convergir para valores coerentes, uma vez que alterando-se a espessura, por exemplo, pode-se encontrar uma força que gere resultados equivalentes, de forma que se cria uma instabilidade matemática no processo de otimização. Assim, foi necessário primeiro manter fixa a geometria e rodar o programa de otimização para encontrar uma força adequada, para somente então permitir que houvesse variação da geometria, mas então mantendo a força fixa. Dessa forma, a otimização teve que ser realizada de forma iterativa, com os resultados sendo cada vez mais refinados.

As extremidades da curva gerada, como se pode ver acima, não apresentaram a mesma qualidade de resultados do que seu centro, e isto se deve ao fato de que, conforme avisado pelo software, houve a criação de uma curva irregular nesses pontos. Para sanar este problema, uma nova maneira de definir a curva foi utilizada.

11.1.3 Coeficientes de um polinômio como variáveis

Uma outra maneira completamente diferente de se definir as variáveis é através dos coeficientes de uma curva qualquer. Para este problema, foi utilizado um polinômio de quarto grau. O número de variáveis foi, assim, muito diminuído, de 11 para apenas 5, sem que se perdesse a flexibilidade da curva gerada. O problema de criação de curvas irregulares apresentado pelo método anterior é assim removido por completo.

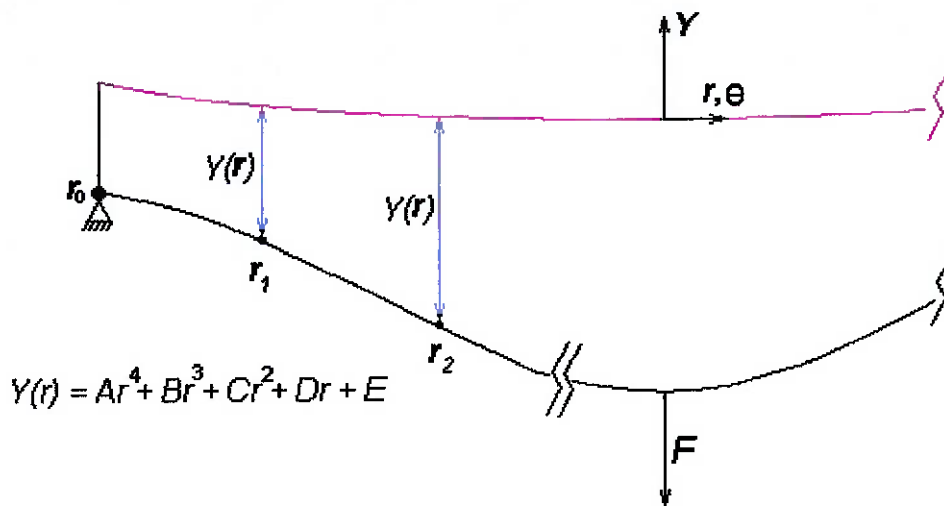


Figura 24. Coeficientes de um polinômio como variáveis

Este método não criou nenhuma curva tão precisa como o método anterior, mas era muito mais simples e confiável. O problema de mínimos locais fora, aparentemente, reduzido, e a partir de quase todas as geometrias iniciais o programa era capaz de criar uma superfície que produzia resultados adequados, dentro da tolerância de 1/10 do comprimento de onda da luz.

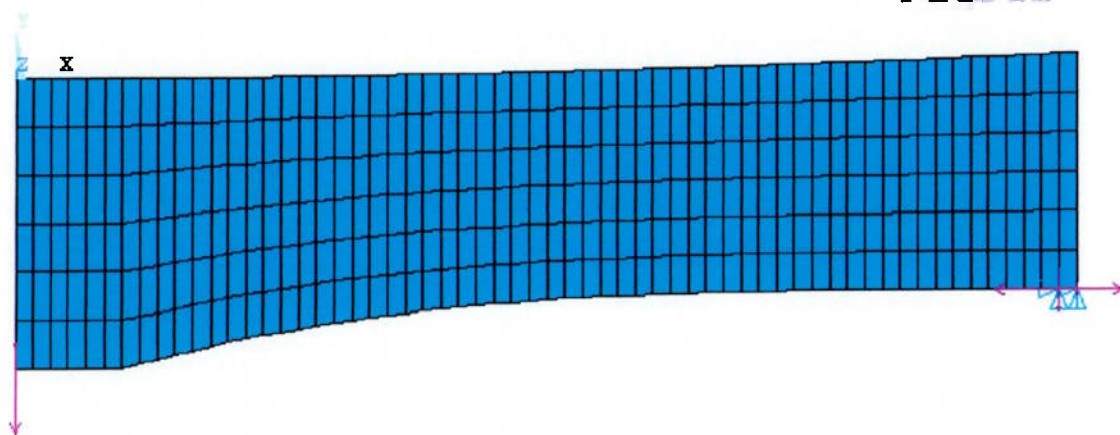


Figura 25. Resultados obtidos

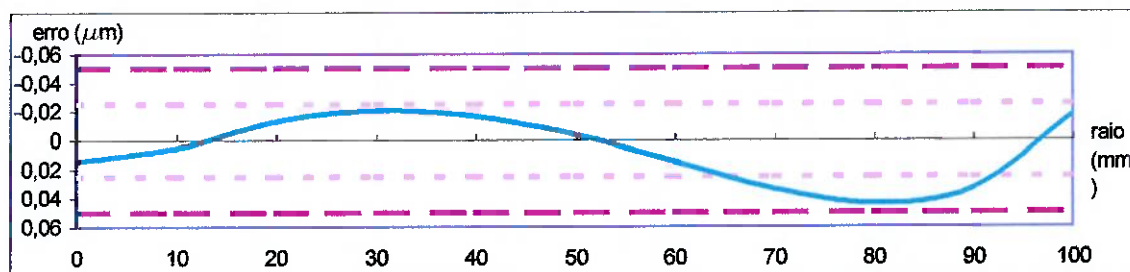


Gráfico 6. Planilha de resultados

Pode-se ver acima como a superfície criada pelo otimizador é bem comportada, não apresentando ângulos abruptos como as superfícies anteriores. A superfície frontal do espelho não apresenta erros maiores do que $1/11$ do comprimento de onda da luz, o que produziria uma imagem de alta qualidade.

11.1.4 Curva de Bézier

A curva de Bézier foi criada por um engenheiro da Renault em 1970, chamado Pierre Bézier, e foi utilizada para a criação do *design* dos carros desta marca. Esta curva é uma *Spline* extremamente bem comportada de implementação relativamente simples.

A curva é dada pela expressão:

$$B(u) = \sum_{k=0}^N P_k \frac{N!}{k!(N-k)!} u^k (1-u)^{(N-k)}$$

Onde P_k é a posição do ponto de controle k , e u varia de zero à um conforme se percorre os pontos de controle. N é o número de pontos de controle menos um.

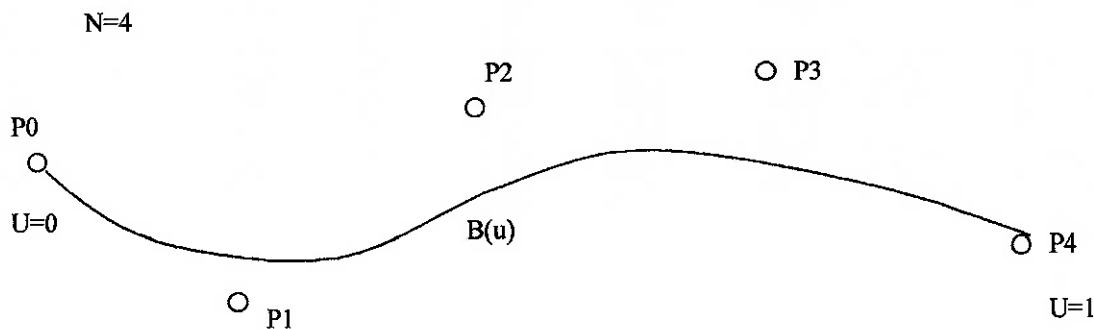


Figura 26. Curva de Bèzier

Na ilustração acima, podemos ver como cinco pontos de controle geram a curva de Bèzier. Esta curva está sempre contida no polígono determinado pelos pontos de controle. Além disso, o valor da curva em suas extremidades é o mesmo que o valor dos pontos de controle, o que equivale a dizer que a curva sempre contém os pontos extremos. A derivada da curva nas extremidades é a mesma do que a do segmento de reta que une o ponto extremo e o ponto adjacente.

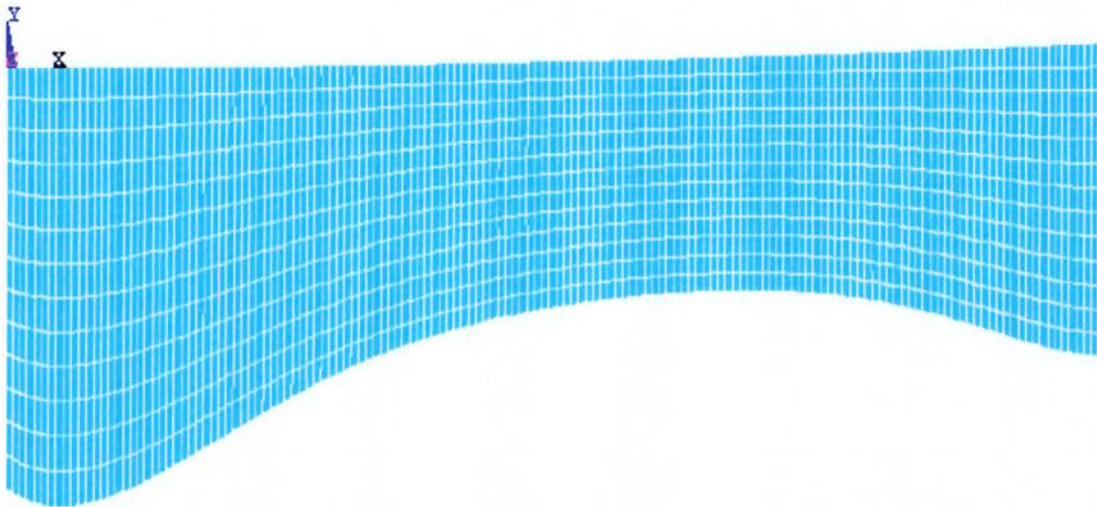


Figura 27. Corte do espelho com curva de Bèzier (eixo de simetria à esquerda)

A otimização utilizando a curva de Bèzier produziu bons resultados, mostrados abaixo:

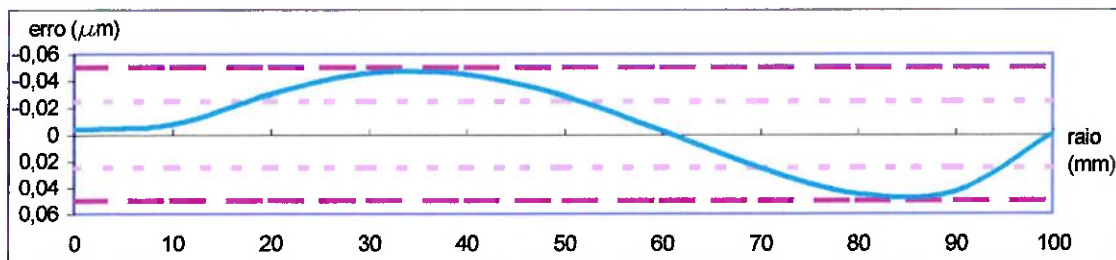


Gráfico 7. Planilhas de resultados

E a geometria que produziu este resultado foi a seguinte:

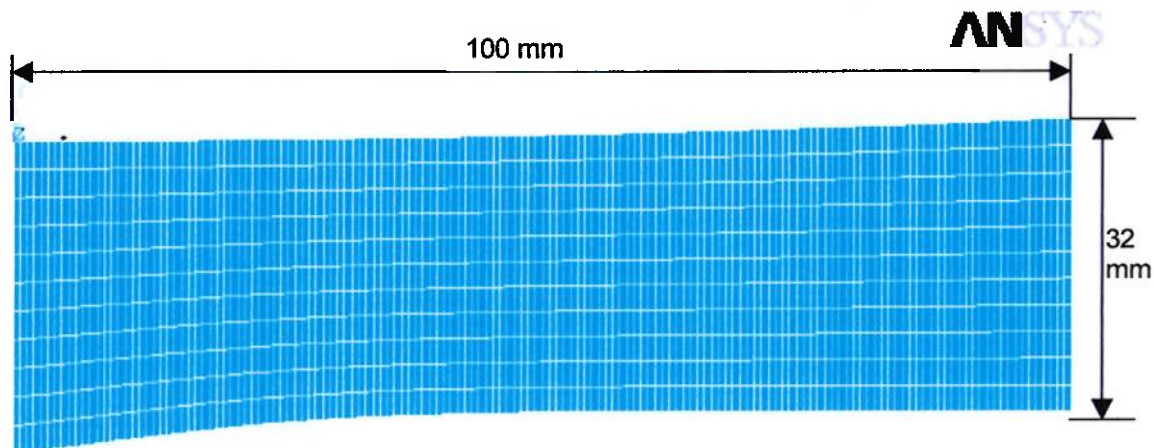


Figura 28. Corte do espelho obtido (eixo de simetria à esquerda)

11.2. Análise de sensibilidade

Uma das principais vantagens do uso de espelhos tensionados é substituir a necessidade de se produzir uma parábola, sendo possível obtê-la a partir de uma superfície esférica. A superfície esférica seria transformada em um parabolóide de revolução a partir da deformação do espelho causada por uma força. O sucesso deste método, porém, depende da forma da superfície traseira do espelho, e isto levanta uma questão: Qual deve ser a precisão desta superfície traseira para que a parábola obtida seja aceitável? Ou, em outras palavras, será que este método simplesmente não transfere a dificuldade de usinagem da superfície frontal para a superfície posterior do espelho?

Para verificar se é este o caso, foi feita uma análise de sensibilidade do modelo. Foram criadas variações aleatórias na geometria produzida ao se utilizar cotas como variáveis (Ver figura 4), simulando erros de usinagem. Essas variações foram da ordem de 0,2 mm, um erro facilmente evitado em máquinas CNC, que seriam utilizadas para produzir a superfície.

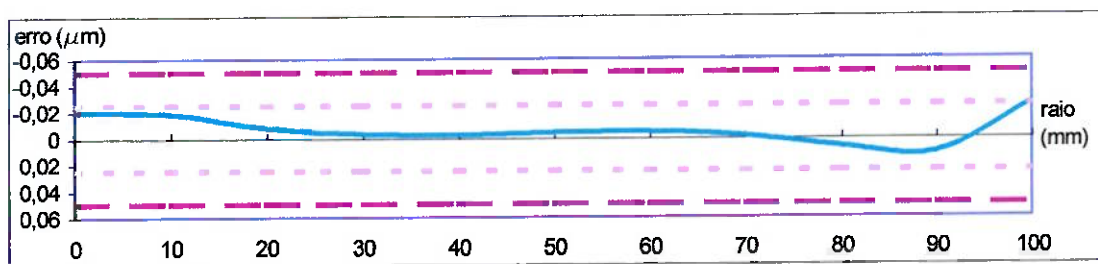


Gráfico 8. Planilhas de resultados

Os resultados obtidos mostram que apesar de ter havido uma mudança na superfície frontal do espelho e um aumento do erro máximo as alterações foram mínimas e a superfície continuou excelente. Assim, conclui-se que a tolerância de usinagem da superfície posterior é grande o suficiente para que o método seja utilizado com sucesso.

12. O Modelo tridimensional

Para a criação do modelo tridimensional, será utilizada uma superfície de Bèzier para definir a superfície inferior do espelho. Esta superfície é obtida ao se combinar curvas de Bèzier em direções diferentes. A expressão da superfície é:

$$B(u, v) = \sum_{i=0}^{N_i} \sum_{j=0}^{N_j} P_{i,j} \frac{N_i!}{i!(N_i-i)!} u^i (1-u)^{(N_i-i)} \frac{N_j!}{j!(N_j-j)!} v^j (1-v)^{(N_j-j)}$$

E é análoga à expressão da curva, mostrada no item 1.4. Assim, u e v variam de 0 à 1 quando se percorrem os pontos de controle em cada uma das direções, e N_i e N_j são sempre o número de pontos de controle na direção respectiva menos 1. Como a curva de Bèzier, a superfície de Bèzier fica sempre contida entre os pontos de controle, e é uma superfície bem-comportada. Nos limites da superfície, esta se reduz a uma curva de Bèzier passando pelos pontos de controle da extremidade da superfície.

O espelho que foi modelado é um espelho para telescópios Cassegrain, por apresentar o furo central que permite a passagem da luz proveniente do espelho secundário. Neste caso, a força não pode ser aplicada no centro do espelho, uma vez que não há centro. A força, assim, é aplicada em três pontos na borda do furo, alinhados com os suportes na periferia do espelho.

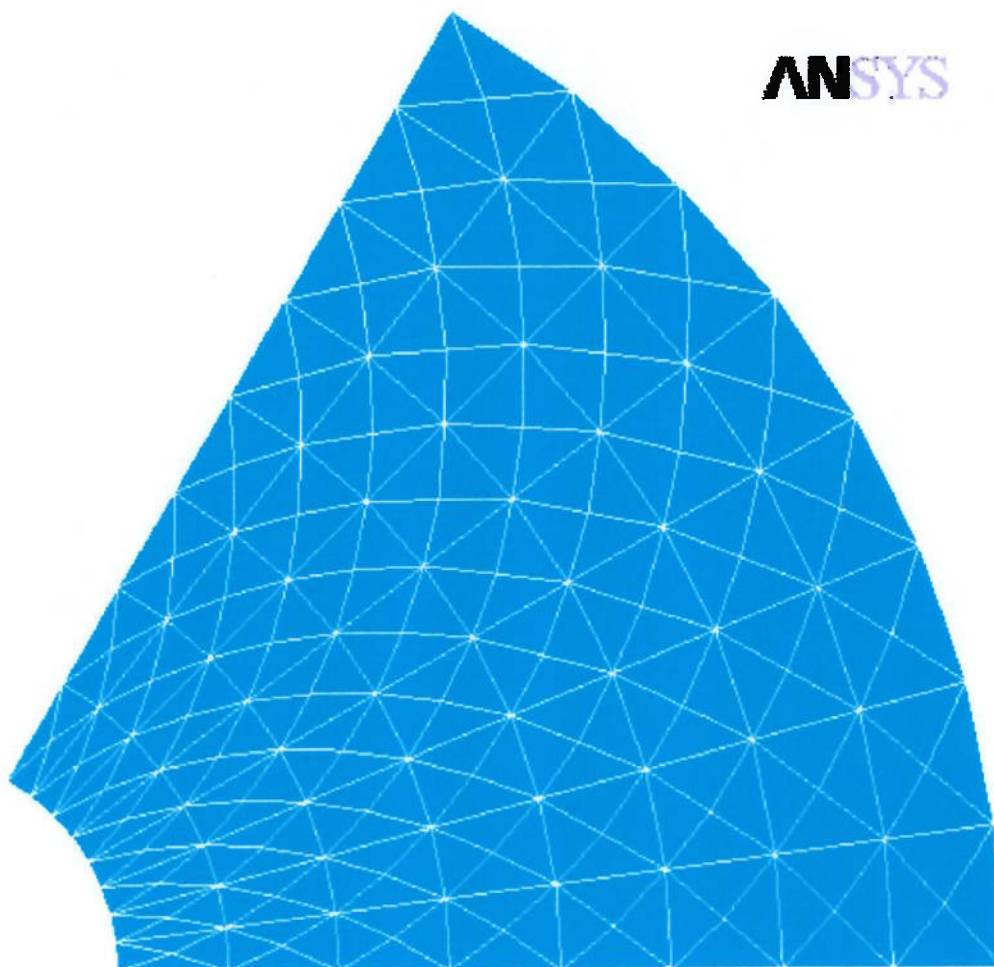


Figura 29. Sistema de coordenadas

O sistema de coordenadas escolhidos foi o cilíndrico, que é o mais adequado ao problema. Somente 1/6 do espelho foi modelado, devido à simetria do problema.

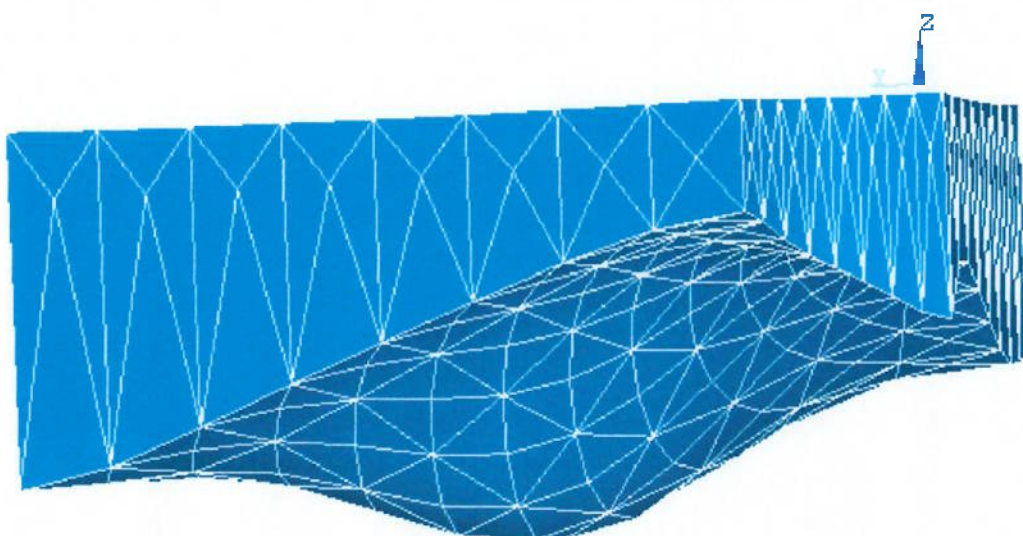


Figura 30. Superfície de Bèzier

Pode-se ver, acima, a superfície de Bèzier implementada. para garantir que o modelo não apresente irregularidades, a derivada da curva foi zerada nas extremidades onde cada "gomo" do espelho toca o adjacente. Apesar de que o conjunto de pontos de controle que gerou a curva era bastante descontínuo e irregular, nota-se que a curva apresenta grande suavidade e não é nem um pouco angulosa, o que contribuiria para erros na otimização.

12.1 Resultados

A diferença entre a superfície do espelho após a aplicação da força e uma parábola de referência está mostrada nos diagramas abaixo. O ponto de aplicação da força é a extremidade inferior direita, e o ponto de apoio do espelho é a extremidade superior direita. A cores do diagrama indicam a distância da superfície obtida em relação a parábola de referência, de acordo com a legenda (em milésimos de milímetro). Note que a tolerância de 1/10 do comprimento de onda da luz equivale a 0,05 milésimos de milímetro.

Uma das principais preocupações no início deste trabalho era o quão adequadamente seria possível restabelecer a simetria da superfície do espelho, uma vez que apoiá-lo em três pontos ao invés de em toda sua periferia romperia a simetria axissimétrica. Podemos observar na figura abaixo que a simetria pode ser quase que completamente restabelecida pela geometria da superfície inferior do espelho, o que valida o método.

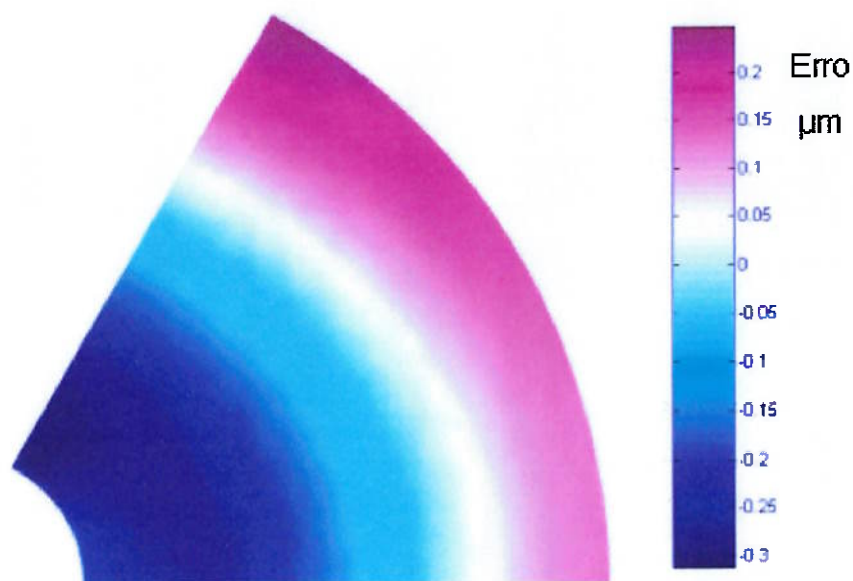


Figura 31. Resultado da otimização

Os resultados obtidos até agora ainda não são plenamente satisfatórios, uma vez que ainda não se conseguiu que a superfície atingisse a precisão necessária, como se pode ver na ilustração acima. Ainda não foi possível, porém, rodar um número suficiente grande de otimizações para que se possa conhecer melhor o domínio do problema e evitar os mínimos locais, de forma que os resultados ainda são muito promissores.

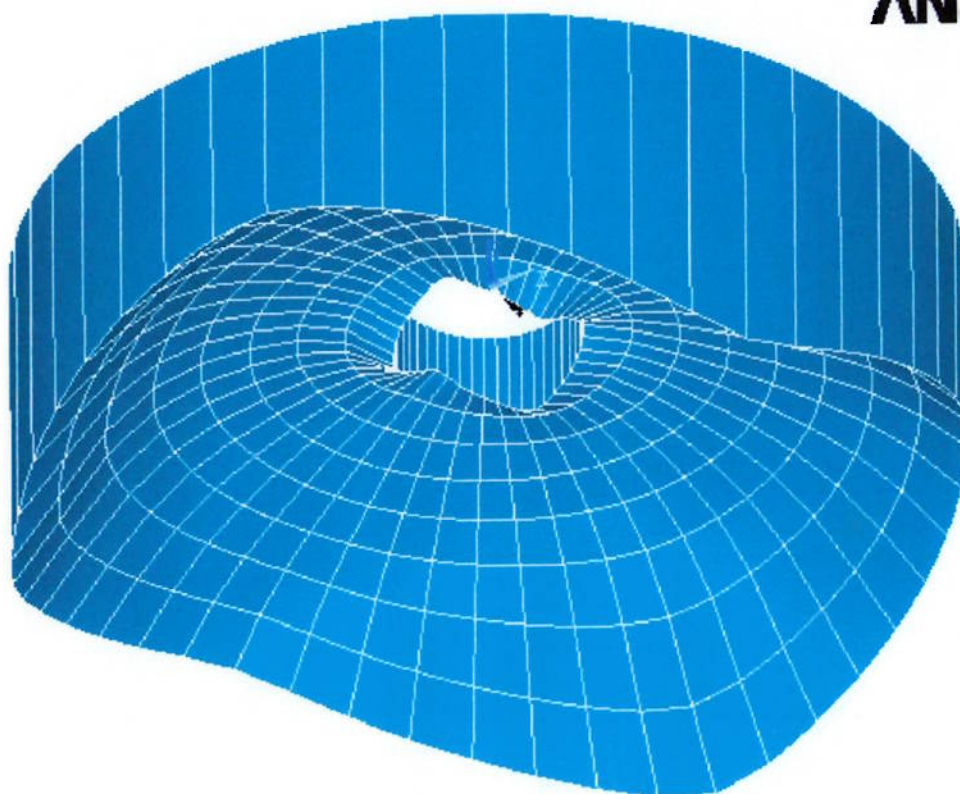


Figura 32. Geometria obtida

13. Comentários e conclusões

Durante a realização deste trabalho, foi possível notar diversas características do processo de otimização, que muitas vezes representaram desafios e aos quais se deve estar atento. As pequenas tolerâncias e deslocamentos muitas vezes criavam más respostas do programa de otimização, e isso somente pode ser resolvido através de um anúncio adequado para a função objetivo. A escolha das variáveis também é um item delicado, uma vez que ao se utilizar variáveis que descreve a geometria do problema e a força ao mesmo tempo, cria-se uma instabilidade matemática no problema, que não produz resultados razoáveis.

Um assunto absolutamente central na otimização são os mínimos locais. O resultado que se obtém a partir de um conjunto de dados de entrada é muito dependente da qualidade destes dados, e a otimização não irá, de forma alguma, obter bons resultados a partir de entradas ruins. Dessa forma, é vital rodar muitas simulações, de forma a conhecer o domínio do problema e ter uma noção intuitiva de como criar uma boa entrada para o problema. É importante, também, ter boas ferramentas de análise das soluções obtidas, não utilizando apenas o valor da função objetivo como parâmetro para se avaliar o desempenho da otimização. Neste trabalho, isto foi feito de forma gráfica, onde se plotou a diferença da superfície do espelho comparada a uma parábola de referência. Dessa forma, pode se ver quais partes do espelho devem ter sua geometria alterada.

De qualquer forma, não foi possível evitar os mínimos locais, o que significa que resultados ainda melhores podem ser obtidos, superando qualquer espelho disponível no mercado atualmente, a um custo menor do que o atual.

Anexo 1. Entrada do programa de otimização para o caso axisimétrico, com diferenças de cotas como variáveis

```
/input,start55,ans ,C:\AnsysED55\docu\,,,,,,,,,,,,,1
/FILNAM,espelho3
/TITLE,espelho3

! Unidades:
! Comprimento: mm
! força: Newton

!!!!!!!!!!!!!!!!!! Construção do modelo !!!!!!!!!!!!!!!!!!!!!!!!!!!!!

/prep7
diametro=200      ! Insira aqui o diâmetro do espelho desejado
R=diametro/2      ! Raio do espelho
razaof=6          ! Insira aqui a razão f do espelho
foco=razaof*diametro ! Distância focal do espelho

Rc=((4*(foco**2))+(R**2))**0.5 !Raio de curvatura da esfera

espess=30
y11=-espess

del10=0
del9=0
del8=0
del7=0
del6=0
del5=0
del4=0
del3=0
del2=0
del1=0
forca=10
f=-forca          ! Força aplicada
```

et,1,plane42,0,0,1 ! Elemento plano axisimétrico

mp,ex,1,703000 ! Propriedades do material

mp,ey,1,703000

mp,ez,1,703000

k,11,R,y11 ! definição de keypoints da sup. inferior

k,10,(9*R/10),(y11-(del10))

k,9,(8*R/10),(y11-(del10+del9))

k,8,(7*R/10),(y11-(del10+del9+del8))

k,7,(6*R/10),(y11-(del7+del10+del9+del8))

k,6,(5*R/10),(y11-(del6+del7+del10+del9+del8))

k,5,(4*R/10),(y11-(del5+del6+del7+del10+del9+del8))

k,4,(3*R/10),(y11-(del4+del5+del6+del7+del10+del9+del8))

k,3,(2*R/10),(y11-(del3+del4+del5+del6+del7+del10+del9+del8))

k,2,(R/10),(y11-(del2+del3+del4+del5+del6+del7+del10+del9+del8))

k,1,0,(y11-(del1+del2+del3+del4+del5+del6+del7+del10+del9+del8))

y10=-(y11-(del10))/50

y9=-(y11-(del10+del9))/50 !Essas serão minhas SVs

y8=-(y11-(del10+del9+del8))/50

y7=-(y11-(del7+del10+del9+del8))/50

y6=-(y11-(del6+del7+del10+del9+del8))/50

y5=-(y11-(del5+del6+del7+del10+del9+del8))/50

y4=-(y11-(del4+del5+del6+del7+del10+del9+del8))/50

y3=-(y11-(del3+del4+del5+del6+del7+del10+del9+del8))/50

y2=-(y11-(del2+del3+del4+del5+del6+del7+del10+del9+del8))/50

y1=-(y11-(del1+del2+del3+del4+del5+del6+del7+del10+del9+del8))/50

! Keypoints superiores

raio=0

Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))

k,12,raio,Er

raio=R/10

Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))

k,13,raio,Er

raio=2*R/10


```

d,3,ux,0,,,uy      ! Deslocamento nulo na periferia
f,29,fy,f          ! Força no centro
solve
finish

```

!!!!!!! Obtenção de dados e enunciado da função objetivo !!!!!!!

```

/post1
set,last
obje=0

```

```

!*get,noux,node,32,u,x      !Noux recebe o deslocamento em X deste n
!*get,nouy,node,32,u,y      !Nouy recebe o deslocamento em Y deste n
!raio=0+noux
!Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
!Pr=((raio**2)/(4*foco))
!aux=nouy+(Er-Pr)
!obje=obje+aux**2

```

```

!*get,noux,node,35,u,x      !Noux recebe o deslocamento em X deste n
!*get,nouy,node,35,u,y      !Nouy recebe o deslocamento em Y deste n
!raio=(1*R/10)+noux
!Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
!Pr=((raio**2)/(4*foco))
!aux=nouy+(Er-Pr)
!obje=obje+aux**2

```

```

*get,noux,node,38,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,38,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(2*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)

```

```

*get,noux,node,41,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,41,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(3*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))

```

```
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2
```

```
*get,noux,node,44,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,44,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(4*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2
```

```
*get,noux,node,47,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,47,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(5*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2
```

```
*get,noux,node,50,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,50,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(6*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2
```

```
*get,noux,node,53,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,53,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(7*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2
```

```

*get,noux,node,56,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,56,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(8*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

```

```

*get,noux,node,59,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,59,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(9*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

```

```

*get,noux,node,62,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,62,u,y      !Nouy recebe o deslocamento em Y deste n
raio=R+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

```

```

finish

```

```

!!!!!!!!!!!!!!!!!!!!!!!!!!!! Otimizador !!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

```

/opt
lgwrite,espelho,lgw,,remove
opanl,espelho,lgw

```

```

opvar,espess,dv,5,50,0.01    !Variáveis de design
opvar,del10,dv,,1,0.01

```

opvar,del9,dv,,1,0.01
opvar,del8,dv,,1,0.01
opvar,del7,dv,,1,0.01
opvar,del6,dv,,1,0.01
opvar,del5,dv,,1,0.01
opvar,del4,dv,,1,0.01
opvar,del3,dv,,1,0.01
opvar,del2,dv,,1,0.01
opvar,del1,dv,,1,0.01
opvar,forca,dv,,100,0.1

opvar,y10,sv,0.2,1 !Variáveis de estado
opvar,y9,sv,0.2,1
opvar,y8,sv,0.2,1
opvar,y7,sv,0.2,1
opvar,y6,sv,0.2,1
opvar,y5,sv,0.2,1
opvar,y4,sv,0.2,1
opvar,y3,sv,0.2,1
opvar,y3,sv,0.2,1
opvar,y1,sv,0.2,1

opvar,obje,obj,,,0.00001 !Função objetivo

opdata,espelho6,opt
optype,first
opfst,10
opexe

finish

Anexo 2. Entrada do programa de otimização para o caso axisimétrico, com cotas como variáveis

```
/input,start55,ans ,C:\AnsysED55\docu\,,,,,,,,,,,,,1
/FILNAM,espelho3
/TITLE,espelho3

! Unidades:
! Comprimento: mm
! força: Newton

!!!!!!!!!!!!!!!!!! Construção do modelo !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

/prep7
diametro=200      ! Insira aqui o diâmetro do espelho desejado
R=diametro/2      ! Raio do espelho
razaof=6          ! Insira aqui a razão f do espelho
foco=razaof*diametro ! Distância focal do espelho

Rc=((4*(foco**2))+(R**2))**0.5      !Raio de curvatura da esfera

y11=21.934
y10=22.181
y9=20.551
y8=20.542
y7=21.024
y6=22.144
y5=24.151
y4=27.367
y3=33.181
y2=45.651
y1= 45.637

forca=3.5699
f=-forca      ! Força aplicada
```

et,1,plane42,0,0,1 ! Elemento plano axisimétrico

mp,ex,1,703000 ! Propriedades do material

mp,ey,1,703000

mp,ez,1,703000

k,11,R,-y11 ! definição de keypoints da sup. inferior

k,10,(9*R/10),-y10

k,9,(8*R/10),-y9

k,8,(7*R/10),-y8

k,7,(6*R/10),-y7

k,6,(5*R/10),-y6

k,5,(4*R/10),-y5

k,4,(3*R/10),-y4

k,3,(2*R/10),-y3

k,2,(R/10),-y2

k,1,0,-y1

! Keypoints superiores

raio=0

$Er = (((Rc^{**2}) - (R^{**2}))^{**0.5}) - (((Rc^{**2}) - (raio^{**2}))^{**0.5}) + ((R^{**2}) / (4 * foco))$

k,12,raio,Er

raio=R/10

$Er = (((Rc^{**2}) - (R^{**2}))^{**0.5}) - (((Rc^{**2}) - (raio^{**2}))^{**0.5}) + ((R^{**2}) / (4 * foco))$

k,13,raio,Er

raio=2*R/10

$Er = (((Rc^{**2}) - (R^{**2}))^{**0.5}) - (((Rc^{**2}) - (raio^{**2}))^{**0.5}) + ((R^{**2}) / (4 * foco))$

k,14,raio,Er

raio=3*R/10

$Er = (((Rc^{**2}) - (R^{**2}))^{**0.5}) - (((Rc^{**2}) - (raio^{**2}))^{**0.5}) + ((R^{**2}) / (4 * foco))$

k,15,raio,Er

raio=4*R/10

$Er = (((Rc^{**2}) - (R^{**2}))^{**0.5}) - (((Rc^{**2}) - (raio^{**2}))^{**0.5}) + ((R^{**2}) / (4 * foco))$

k,16,raio,Er

```

raio=5*R/10
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,17,raio,Er
raio=6*R/10
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,18,raio,Er
raio=7*R/10
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,19,raio,Er
raio=8*R/10
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,20,raio,Er
raio=9*R/10
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,21,raio,Er
raio=R
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,22,raio,Er

```

```

a,12,13,14,15,16,17,6,5,4,3,2,1    ! Uma área
a,17,18,19,20,21,22,11,10,9,8,7,6  ! Outra área
aadd,1,2                            ! Soma das áreas
mshape0,2d                          ! propriedades da malha
mshkey,1
amap,3,1,12,22,11                  ! Criar a malha
finish

```

!!!!!!!!!!!!!!!!!!!!!!!!!!!! Obtenção da Solução !!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

/solu
d,1,ux,0,,,uy
d,3,ux,0,,,uy    ! Deslocamento nulo na periferia
f,29,fy,f        ! Força no centro
solve
finish

```

!!!!!!!!!!!! Obtenção de dados e enunciado da função objetivo !!!!!!!!!!

/post1
set,last
obje=0

!*get,noux,node,32,u,x !Noux recebe o deslocamento em X deste n
!*get,nouy,node,32,u,y !Nouy recebe o deslocamento em Y deste n
!raio=0+noux
!Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
!Pr=((raio**2)/(4*foco))
!aux=nouy+(Er-Pr)
!obje=obje+aux**2

!*get,noux,node,35,u,x !Noux recebe o deslocamento em X deste n
!*get,nouy,node,35,u,y !Nouy recebe o deslocamento em Y deste n
!raio=(1*R/10)+noux
!Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
!Pr=((raio**2)/(4*foco))
!aux=nouy+(Er-Pr)
!obje=obje+aux**2

*get,noux,node,38,u,x !Noux recebe o deslocamento em X deste n
*get,nouy,node,38,u,y !Nouy recebe o deslocamento em Y deste n
raio=(2*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

*get,noux,node,41,u,x !Noux recebe o deslocamento em X deste n
*get,nouy,node,41,u,y !Nouy recebe o deslocamento em Y deste n
raio=(3*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

*get,noux,node,44,u,x !Noux recebe o deslocamento em X deste n
*get,nouy,node,44,u,y !Nouy recebe o deslocamento em Y deste n
raio=(4*R/10)+noux


```

Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

```

```

*get,noux,node,47,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,47,u,y     !Nouy recebe o deslocamento em Y deste n
raio=(5*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

```

```

*get,noux,node,50,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,50,u,y     !Nouy recebe o deslocamento em Y deste n
raio=(6*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

```

```

*get,noux,node,53,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,53,u,y     !Nouy recebe o deslocamento em Y deste n
raio=(7*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

```

```

*get,noux,node,56,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,56,u,y     !Nouy recebe o deslocamento em Y deste n
raio=(8*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

```

```

*get,noux,node,59,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,59,u,y     !Nouy recebe o deslocamento em Y deste n
raio=(9*R/10)+noux

```

```

Er=(((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

```

```

*get,noux,node,62,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,62,u,y      !Nouy recebe o deslocamento em Y deste n
raio=R+noux
Er=(((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

```

```

finish

```

```

!!!!!!!!!!!!!!!!!!!!!!!!!!!! Otimizador !!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

```

/opt
lgwrite,espelho,lgw,,remove
opanl,espelho,lgw

```

```

opvar,y11,dv,5,50,      !Variáveis de design
opvar,y10,dv,5,50,
opvar,y9,dv,5,50,
opvar,y8,dv,5,50,
opvar,y7,dv,5,50,
opvar,y6,dv,5,50,
opvar,y5,dv,5,50,
opvar,y4,dv,5,50,
opvar,y3,dv,5,50,
opvar,y2,dv,5,50,
opvar,y1,dv,5,50,
!opvar,forca,dv,,100,

```

```

opvar,obje,obj,,,10E-20      !Função objetivo

```

```

!optype,rand
!oprnd,10,0
!opexe

```

!opsave,espelhorand,opt
!opsel,-1

opdata,espelho6,opt
optype,first
opfrst,20
opexe

!/axlab,y,objetivo
!plvar,obje

!/axlab,y,força,forca
!plvar,forca

!/axlab,y,espessura,espess
!plvar,espess

finish

Anexo 3. Entrada do programa de otimização para o caso axisimétrico, com coeficientes de um polinômio como variáveis

```
/input,start55,ans ,C:\AnsysED55\docu\,,,,,,,,,,,,,1
/FILNAM,espelho3
/TITLE,espelho3

! Unidades:
! Comprimento: mm
! força: Newton

!!!!!!!!!!!!!! Construção do modelo !!!!!!!!!!!!!!!

/prep7
diametro=200      ! Insira aqui o diâmetro do espelho desejado
R=diametro/2      ! Raio do espelho
razaof=6          ! Insira aqui a razão f do espelho
foco=razaof*diametro ! Distância focal do espelho

Rc=((4*(foco**2))+(R**2))**0.5 !Raio de curvatura da esfera

a=54.181
b=58.589
c=47.568
d=50.030
e=20.073
forca= 2.1771
f=-forca          ! Força aplicada

et,1,plane42,0,0,1 ! Elemento plano axisimétrico

mp,ex,1,703000     ! Propriedades do material
mp,ey,1,703000
mp,ez,1,703000

! definição de keypoints da sup. inferior
```

```

ra=R
t1=((a-50)/(10E7))*((100-ra)**4)
t2=((b-50)/(10E5))*((100-ra)**3)
t3=((c-50)/(10E3))*((100-ra)**2)
t4=((d-50)/(10E2))*(100-ra)
sup=t1+t2+t3+t4+e
k,11,ra,0-sup

```

```

ra=(9*R/10)
t1=((a-50)/(10E7))*((100-ra)**4)
t2=((b-50)/(10E5))*((100-ra)**3)
t3=((c-50)/(10E3))*((100-ra)**2)
t4=((d-50)/(10E2))*(100-ra)
sup=t1+t2+t3+t4+e
k,10,ra,0-sup

```

```

ra=(8*R/10)
t1=((a-50)/(10E7))*((100-ra)**4)
t2=((b-50)/(10E5))*((100-ra)**3)
t3=((c-50)/(10E3))*((100-ra)**2)
t4=((d-50)/(10E2))*(100-ra)
sup=t1+t2+t3+t4+e
k,9,ra,0-sup

```

```

ra=(7*R/10)
t1=((a-50)/(10E7))*((100-ra)**4)
t2=((b-50)/(10E5))*((100-ra)**3)
t3=((c-50)/(10E3))*((100-ra)**2)
t4=((d-50)/(10E2))*(100-ra)
sup=t1+t2+t3+t4+e
k,8,ra,0-sup

```

```

ra=(6*R/10)
t1=((a-50)/(10E7))*((100-ra)**4)
t2=((b-50)/(10E5))*((100-ra)**3)
t3=((c-50)/(10E3))*((100-ra)**2)
t4=((d-50)/(10E2))*(100-ra)
sup=t1+t2+t3+t4+e
k,7,ra,0-sup

```

```

ra=(5*R/10)
t1=((a-50)/(10E7))*((100-ra)**4)
t2=((b-50)/(10E5))*((100-ra)**3)
t3=((c-50)/(10E3))*((100-ra)**2)
t4=((d-50)/(10E2))*(100-ra)
sup=t1+t2+t3+t4+e
k,6,ra,0-sup

```

```

ra=(4*R/10)
t1=((a-50)/(10E7))*((100-ra)**4)
t2=((b-50)/(10E5))*((100-ra)**3)
t3=((c-50)/(10E3))*((100-ra)**2)
t4=((d-50)/(10E2))*(100-ra)
sup=t1+t2+t3+t4+e
k,5,ra,0-sup

```

```

ra=(3*R/10)
t1=((a-50)/(10E7))*((100-ra)**4)
t2=((b-50)/(10E5))*((100-ra)**3)
t3=((c-50)/(10E3))*((100-ra)**2)
t4=((d-50)/(10E2))*(100-ra)
sup=t1+t2+t3+t4+e
k,4,ra,0-sup

```

```

ra=(2*R/10)
t1=((a-50)/(10E7))*((100-ra)**4)
t2=((b-50)/(10E5))*((100-ra)**3)
t3=((c-50)/(10E3))*((100-ra)**2)
t4=((d-50)/(10E2))*(100-ra)
sup=t1+t2+t3+t4+e
k,3,ra,0-sup

```

```

ra=(R/10)
t1=((a-50)/(10E7))*((100-ra)**4)
t2=((b-50)/(10E5))*((100-ra)**3)
t3=((c-50)/(10E3))*((100-ra)**2)
t4=((d-50)/(10E2))*(100-ra)
sup=t1+t2+t3+t4+e
k,2,ra,0-sup

```

```

ra=0
t1=((a-50)/(10E7))*((100-ra)**4)
t2=((b-50)/(10E5))*((100-ra)**3)
t3=((c-50)/(10E3))*((100-ra)**2)
t4=((d-50)/(10E2))*(100-ra)
!sup=t1+t2+t3+t4+e
k,1,ra,0-sup

```

```

supmax=sup

```

! Keypoints superiores

```

raio=0
Er=(((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,12,raio,Er
raio=R/10
Er=(((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,13,raio,Er
raio=2*R/10
Er=(((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,14,raio,Er
raio=3*R/10
Er=(((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,15,raio,Er
raio=4*R/10
Er=(((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,16,raio,Er
raio=5*R/10
Er=(((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,17,raio,Er
raio=6*R/10
Er=(((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,18,raio,Er
raio=7*R/10
Er=(((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,19,raio,Er
raio=8*R/10
Er=(((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,20,raio,Er
raio=9*R/10

```

```

Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,21,raio,Er
raio=R
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
k,22,raio,Er

```

```

a,12,13,14,15,16,17,6,5,4,3,2,1    ! Uma área
a,17,18,19,20,21,22,11,10,9,8,7,6  ! Outra área
aadd,1,2                            ! Soma das áreas
mshape,0,2d                         ! propriedades da malha
mshkey,1
amap,3,1,12,22,11                  ! Criar a malha
erefine,all,,1,0,1,1
finish

```

!!!!!!!!!!!!!!!!!!!!!!!!!!!! Obtenção da Solução !!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

/solu
d,1,ux,0,,,uy
d,3,ux,0,,,uy                      ! Deslocamento nulo na periferia
f,29,fy,f                          ! Força no centro
solve
finish

```

!!!!!!!!!!!! Obtenção de dados e enunciado da função objetivo !!!!!!!!!

```

/post1
set,last
obje=0

```

```

*get,noux,node,32,u,x              !Noux recebe o deslocamento em X deste n
*get,nouy,node,32,u,y              !Nouy recebe o deslocamento em Y deste n
raio=0+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

```

```

*get,noux,node,35,u,x              !Noux recebe o deslocamento em X deste n
*get,nouy,node,35,u,y              !Nouy recebe o deslocamento em Y deste n

```



```

raio=(1*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

*get,noux,node,38,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,38,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(2*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

*get,noux,node,41,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,41,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(3*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

*get,noux,node,44,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,44,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(4*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

*get,noux,node,47,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,47,u,y      !Nouy recebe o deslocamento em Y deste n
raio=(5*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

*get,noux,node,50,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,50,u,y      !Nouy recebe o deslocamento em Y deste n

```

```

raio=(6*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

*get,noux,node,53,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,53,u,y     !Nouy recebe o deslocamento em Y deste n
raio=(7*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

*get,noux,node,56,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,56,u,y     !Nouy recebe o deslocamento em Y deste n
raio=(8*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

*get,noux,node,59,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,59,u,y     !Nouy recebe o deslocamento em Y deste n
raio=(9*R/10)+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

*get,noux,node,62,u,x      !Noux recebe o deslocamento em X deste n
*get,nouy,node,62,u,y     !Nouy recebe o deslocamento em Y deste n
raio=R+noux
Er=((((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))
Pr=((raio**2)/(4*foco))
aux=nouy+(Er-Pr)
obje=obje+aux**2

finish

```

!!!!!!!!!!!!!!!!!!!!!! Otimizador !!!!!!!!!!!!!!!!!!!!!!!

/opt

lgwrite,espelho,lgw,,remove

opani,espelho,lgw

opvar,a,dv,,120, !Variáveis de design

opvar,b,dv,,120,

opvar,c,dv,,120,

opvar,d,dv,,120,

opvar,e,dv,,30,

!opvar,forca,dv,,100,

!opvar,supmax,sv,10,50

opvar,obje,obj,,10E-15 !Função objetivo

loptype,rand

loprand,10,0

lopexe

lopsave,espelhorand,opt

loysel,-1

opdata,espelho6,opt

optype,first

opfrst,10

opexe

!/axlab,y,objetivo

!plvar,obje

!/axlab,y,força,forca

!plvar,forca

!/axlab,y,espessura,espess

!plvar,espess

finish

Anexo 4. Programa em MatLab que escreve a entrada do programa de otimização para o caso axisimétrico, com coeficientes de uma curva de Bèzier como variáveis

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   André Izecon de Carvalho
%   Escola Politécnica da USP
%
%   Trabalho de formatura 2002
%   Projeto de um novo tipo de espelho para telescópios
%   Caso axisimétrico-Parabólico
%   Curva de Bèzier
%
%   Orientador: Prof. Dr. Emílio Carlos Nelli Silva
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

outfile=input('Output file: ','s');
fid=fopen(outfile,'w');

fprintf(fid,'%s\n','/FILNAM,bz2d');
fprintf(fid,'%s\n','/TITLE,bz2d');

fprintf(fid,'%s\n','/prep7');

fprintf(fid,'%s\n','f=4');
fprintf(fid,'%s\n','forca=-f');

fprintf(fid,'%s\n','espess=0');

fprintf(fid,'%s\n','y0=38.2');
fprintf(fid,'%s\n','y1=45.50');
fprintf(fid,'%s\n','y2=32.27');
fprintf(fid,'%s\n','y3=23.69');
fprintf(fid,'%s\n','y4=21.05');
fprintf(fid,'%s\n','y5=21.24');
fprintf(fid,'%s\n','y6=20.76');
fprintf(fid,'%s\n','y7=18.30');
fprintf(fid,'%s\n','y8=17.42');
fprintf(fid,'%s\n','y9=26.41');
fprintf(fid,'%s\n','y10=26.40');

fprintf(fid,'%s\n','et,1,plane42,0,0,1 ');
fprintf(fid,'%s\n','mp,ex,1,730 ');
fprintf(fid,'%s\n','mp,ey,1,730 ');
fprintf(fid,'%s\n','mp,ez,1,730 ');

diametro=200;           %Insira aqui o diâmetro do espelho
desejado
R=diametro/2;           %Raio do espelho
razaof=6;               %Insira aqui a razão f do espelho
foco=razaof*diametro;   %Distância focal do espelho
VG=', ';                %String "Virgula"

```

```

kp=100;
Rc=((4*(foco^2))+(R^2))^0.5;

focostr=num2str(foco);
comando=strcat('foco=',focostr);
fprintf(fid,'%s\n',comando);

Rstr=num2str(R);
comando=strcat('R=',Rstr);
fprintf(fid,'%s\n',comando);

Rcstr=num2str(Rc);
comando=strcat('Rc=',Rcstr);
fprintf(fid,'%s\n',comando);


raio=0;

while raio<101
    Er=((Rc^2)-(R^2))^0.5-(((Rc^2)-(raio^2))^0.5)+((R^2)/(4*foco));
    Erstr=num2str(Er);
    raiostr=num2str(raio);
    kpstr=num2str(kp);
    comando=strcat('K,',kpstr,VG,raiostr,VG,Erstr);
    fprintf(fid,'%s\n',comando);

    raio=raio+2;
    kp=kp+1;
end

raio=0;
kp=200;
Ni=10;

while raio<101
    u=raio/100;
    i=0;
    fprintf(fid,'%s\n','bezier=0');
    while i<Ni+1
        aux1=factorial(Ni)/((factorial(i))*(factorial(Ni-i)));
        aux2=(u^i)*((1-u)^(Ni-i));
        blend=aux1*aux2;
        istr=num2str(i);
        blendstr=num2str(blend);
        comando=strcat('Pi=y',istr);
        fprintf(fid,'%s\n',comando);
        comando=strcat('blendstr=',blendstr);
        fprintf(fid,'%s\n',comando);
        fprintf(fid,'%s\n','bezier=bezier+((-Pi-espess)*blendstr)');

        i=i+1;
    end

    raiostr=num2str(raio);
    kpstr=num2str(kp);
    bezier='bezier';
    comando=strcat('K,',kpstr,VG,raiostr,VG,bezier);
    fprintf(fid,'%s\n',comando);

```

```

    raio=raio+2;
    kp=kp+1;
end

cont=100;

while cont<150
    kp1=cont;
    kp2=cont+1;
    kp3=cont+101;
    kp4=cont+100;
    kp1str=num2str(kp1);
    kp2str=num2str(kp2);
    kp3str=num2str(kp3);
    kp4str=num2str(kp4);
    comando=strcat('A,',kp1str,VG,kp2str,VG,kp3str,VG,kp4str);
    fprintf(fid,'%s\n',comando);

    cont=cont+1;
end

```

```

fprintf(fid,'%s\n','Asel,,area,,1,50');
fprintf(fid,'%s\n','aadd,all');
fprintf(fid,'%s\n','mshape,0,2d');
fprintf(fid,'%s\n','mshkey,1');
fprintf(fid,'%s\n','amap,51,100,150,200,250');
fprintf(fid,'%s\n','finish ');

```

```

fprintf(fid,'%s\n','/solu');
fprintf(fid,'%s\n','d,1,ux,0,,,uy');
fprintf(fid,'%s\n','d,2,ux,0,,,uy');
fprintf(fid,'%s\n','f,149,fy,forca');
fprintf(fid,'%s\n','solve');
fprintf(fid,'%s\n','finish');

```

```

fprintf(fid,'%s\n','/post1 ');
fprintf(fid,'%s\n','set,last');
fprintf(fid,'%s\n','obje=0');

```

```

no=152;
raio=0;
nostr=num2str(no);
raiostr=num2str(raio);
ux='u,x';
uy='u,y';
maisnoux='+noux';
comando=strcat('*get,noux,node,',nostr,VG,ux);
fprintf(fid,'%s\n',comando);
comando=strcat('*get,nouy,node,',nostr,VG,uy);
fprintf(fid,'%s\n',comando);
comando=strcat('raio=',raiostr,maisnoux);
fprintf(fid,'%s\n',comando);
fprintf(fid,'%s\n','Er=((Rc**2)-(R**2))**0.5)-(((Rc**2)-(raio**2))**0.5)+((R**2)/(4*foco))');

```

```

fprintf(fid,'%s\n','Pr=((raio**2)/(4*foco))');
fprintf(fid,'%s\n','aux=nouy+(Er-Pr)');
fprintf(fid,'%s\n','obje=obje+aux**2');

no=160;
raio=2;

while no<308
    nostr=num2str(no);
    raiostr=num2str(raio);
    ux='u,x';
    uy='u,y';
    maisnoux='+noux';
    comando=strcat('*get,noux,node,',nostr,VG,ux);
    fprintf(fid,'%s\n',comando);
    comando=strcat('*get,nouy,node,',nostr,VG,uy);
    fprintf(fid,'%s\n',comando);
    comando=strcat('raio=',raiostr,maisnoux);
    fprintf(fid,'%s\n',comando);
    fprintf(fid,'%s\n','Er=((Rc**2)-(R**2))*0.5)-(((Rc**2)-
(radio**2))*0.5)+((R**2)/(4*foco))');
    fprintf(fid,'%s\n','Pr=((raio**2)/(4*foco))');
    fprintf(fid,'%s\n','aux=nouy+(Er-Pr)');
    fprintf(fid,'%s\n','obje=obje+aux**2');

    no=no+3;
    raio=raio+2;

```

end

```

fprintf(fid,'%s\n','obje=obje*10000000');
fprintf(fid,'%s\n','finish');

fprintf(fid,'%s\n','/opt');
fprintf(fid,'%s\n','lgwrite,espelho,lgw,,remove');
fprintf(fid,'%s\n','opanl,espelho,lgw');
fprintf(fid,'%s\n','opvar,y0,dv,5,50,');
fprintf(fid,'%s\n','opvar,y1,dv,5,50,');
fprintf(fid,'%s\n','opvar,y2,dv,5,50,');
fprintf(fid,'%s\n','opvar,y3,dv,5,50,');
fprintf(fid,'%s\n','opvar,y4,dv,5,50,');
fprintf(fid,'%s\n','opvar,y5,dv,5,50,');
fprintf(fid,'%s\n','opvar,y6,dv,5,50,');
fprintf(fid,'%s\n','opvar,y7,dv,5,50,');
fprintf(fid,'%s\n','opvar,y8,dv,5,50,');
fprintf(fid,'%s\n','opvar,y9,dv,5,50,');
fprintf(fid,'%s\n','opvar,y10,dv,5,50,');

fprintf(fid,'%s\n','opvar,f,dv,,100,');

fprintf(fid,'%s\n','opvar,obje,obj,,10E-20');

fprintf(fid,'%s\n','opdata,espelho6,opt');
fprintf(fid,'%s\n','optype,first');
fprintf(fid,'%s\n','opfirst,5');
fprintf(fid,'%s\n','opexe');
fprintf(fid,'%s\n','finish');

```

fclose(fid);

clear all;

Anexo 5. Programa em MatLab que escreve a entrada do programa de otimização para o caso tridimensional, com coeficientes de uma superfície de Bèzier como variáveis

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   André Izecon de Carvalho
%   Escola Politécnica da USP
%
%   Trabalho de formatura 2002
%   Projeto de um novo tipo de espelho para telescópios
%   Caso Tridimensional / Parabólico
%   Curva de Bèzier
%
%   Orientador: Prof. Dr. Emílio Carlos Nelli Silva
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

outfile=input('Output file: ','s');
fid=fopen(outfile,'w');

fprintf(fid,'%s\n','!/input,start55,ans
,C:\AnsysED55\docu\,,,,,,,,,,,,,1');
fprintf(fid,'%s\n','/FILNAM,tridim');
fprintf(fid,'%s\n','/TITLE,tridim');

fprintf(fid,'%s\n','/prep7');

fprintf(fid,'%s\n','forca=100');

fprintf(fid,'%s\n','espress=20');
fprintf(fid,'%s\n','menoses=-espress');
menoses='menoses';

fprintf(fid,'%s\n','y00=0');
fprintf(fid,'%s\n','y01=y00');
fprintf(fid,'%s\n','y02=0');
fprintf(fid,'%s\n','y03=0');
fprintf(fid,'%s\n','y04=0');
fprintf(fid,'%s\n','y05=0');
fprintf(fid,'%s\n','y06=0');
fprintf(fid,'%s\n','y07=0');
fprintf(fid,'%s\n','y08=y07');

fprintf(fid,'%s\n','y10=0');
fprintf(fid,'%s\n','y11=y10');
fprintf(fid,'%s\n','y12=0');
fprintf(fid,'%s\n','y13=0');
fprintf(fid,'%s\n','y14=0');
fprintf(fid,'%s\n','y15=0');
fprintf(fid,'%s\n','y16=0');
fprintf(fid,'%s\n','y17=0');
fprintf(fid,'%s\n','y18=y17');

fprintf(fid,'%s\n','y20=0');
fprintf(fid,'%s\n','y21=y20');
fprintf(fid,'%s\n','y22=0');
fprintf(fid,'%s\n','y23=0');

```



```
fprintf(fid, '%s\n', 'y24=0');
fprintf(fid, '%s\n', 'y25=0');
fprintf(fid, '%s\n', 'y26=0');
fprintf(fid, '%s\n', 'y27=0');
fprintf(fid, '%s\n', 'y28=y27');
```

```
fprintf(fid, '%s\n', 'y30=0');
fprintf(fid, '%s\n', 'y31=y30');
fprintf(fid, '%s\n', 'y32=0');
fprintf(fid, '%s\n', 'y33=0');
fprintf(fid, '%s\n', 'y34=0');
fprintf(fid, '%s\n', 'y35=0');
fprintf(fid, '%s\n', 'y36=0');
fprintf(fid, '%s\n', 'y37=0');
fprintf(fid, '%s\n', 'y38=y37');
```

```
fprintf(fid, '%s\n', 'y40=0');
fprintf(fid, '%s\n', 'y41=y40');
fprintf(fid, '%s\n', 'y42=0');
fprintf(fid, '%s\n', 'y43=0');
fprintf(fid, '%s\n', 'y44=0');
fprintf(fid, '%s\n', 'y45=0');
fprintf(fid, '%s\n', 'y46=0');
fprintf(fid, '%s\n', 'y47=0');
fprintf(fid, '%s\n', 'y48=y47');
```

```
fprintf(fid, '%s\n', 'y50=0');
fprintf(fid, '%s\n', 'y51=y50');
fprintf(fid, '%s\n', 'y52=0');
fprintf(fid, '%s\n', 'y53=0');
fprintf(fid, '%s\n', 'y54=0');
fprintf(fid, '%s\n', 'y55=0');
fprintf(fid, '%s\n', 'y56=0');
fprintf(fid, '%s\n', 'y57=0');
fprintf(fid, '%s\n', 'y58=y57');
```

```
fprintf(fid, '%s\n', 'y60=0');
fprintf(fid, '%s\n', 'y61=y60');
fprintf(fid, '%s\n', 'y62=0');
fprintf(fid, '%s\n', 'y63=0');
fprintf(fid, '%s\n', 'y64=0');
fprintf(fid, '%s\n', 'y65=0');
fprintf(fid, '%s\n', 'y66=0');
fprintf(fid, '%s\n', 'y67=0');
fprintf(fid, '%s\n', 'y68=y67');
```

```
fprintf(fid, '%s\n', 'y70=0');
fprintf(fid, '%s\n', 'y71=y70');
fprintf(fid, '%s\n', 'y72=0');
fprintf(fid, '%s\n', 'y73=0');
fprintf(fid, '%s\n', 'y74=0');
fprintf(fid, '%s\n', 'y75=0');
fprintf(fid, '%s\n', 'y76=0');
fprintf(fid, '%s\n', 'y77=0');
fprintf(fid, '%s\n', 'y78=y77');
```

```
fprintf(fid, '%s\n', 'y80=0');
fprintf(fid, '%s\n', 'y81=y80');
fprintf(fid, '%s\n', 'y82=0');
fprintf(fid, '%s\n', 'y83=0');
fprintf(fid, '%s\n', 'y84=0');
fprintf(fid, '%s\n', 'y85=0');
fprintf(fid, '%s\n', 'y86=0');
fprintf(fid, '%s\n', 'y87=0');
```

```

fprintf(fid, '%s\n' , 'y88=y87');

fprintf(fid, '%s\n' , 'MP,EX,1,703000');
fprintf(fid, '%s\n' , 'MP,EY,1,703000');
fprintf(fid, '%s\n' , 'MP,EZ,1,703000');

fprintf(fid, '%s\n' , 'csys,1');

r=0;
theta=0;
pi=3.14159299;
raio=0;
kp=1;                                %número do keypoint
diametro=200;                         %Insira aqui o diâmetro do espelho
desejado
R=diametro/2;                         %Raio do espelho
razaof=6;                             %Insira aqui a razão f do espelho
foco=razaof*diametro;                 %Distância focal do espelho
VG=', ';                             %String "Vírgula"

Rc=((4*(foco^2))+(R^2))^0.5;

Rcstr=num2str(Rc);
comando=strcat('Rc=', Rcstr);
fprintf(fid, '%s\n' , comando);
FOCOstr=num2str(foco);
comando=strcat('foco=', FOCOstr);
fprintf(fid, '%s\n' , comando);
Rstr=num2str(R);
comando=strcat('R=', Rstr);
fprintf(fid, '%s\n' , comando);

Er=((Rc^2)-(R^2))^0.5-(((Rc^2)-(raio^2))^0.5)+((R^2)/(4*foco));
Erstr=num2str(Er);
rstr=num2str(r);
thetastr=num2str(theta);

raio=20;
conttheta=0;

while raio<101
    raiostr=num2str(raio);
    while conttheta<9

        theta=conttheta*7.5;

        %%%%%%%%% Superficie superior %%%%%%%%%

        Er=((Rc^2)-(R^2))^0.5-(((Rc^2)-(raio^2))^0.5)+((R^2)/(4*foco));

        Erstr=num2str(Er);
        thetastr=num2str(theta);
        kpstr=num2str(kp);

```

```

        comando=strcat('K,',kpstr,VG,raiostr,VG,thetastr,VG,Erstr);
        fprintf(fid,'%s\n',comando);
        kp=kp+1000;
        kpstr=num2str(kp);
        comando=strcat('K,',kpstr,VG,raiostr,VG,thetastr,VG,menoses);
        fprintf(fid,'%s\n',comando);
        kp=kp-1000;

        kp=kp+1;
        conttheta=conttheta+1;
    end

    conttheta=0;
    raio=raio+10;
end

cont1=1;
cont2=1;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%>>--  DEFINIÇÃO DOS VOLUMES  --<<:%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
while cont1<72
    while cont2<9
        kp1=cont1;
        kp2=cont1+1;
        kp3=cont1+9;
        kp4=cont1+10;
        kp5=kp1+1000;
        kp6=kp2+1000;
        kp7=kp3+1000;
        kp8=kp4+1000;
        kp1str=num2str(kp1);
        kp2str=num2str(kp2);
        kp3str=num2str(kp3);
        kp4str=num2str(kp4);
        kp5str=num2str(kp5);
        kp6str=num2str(kp6);
        kp7str=num2str(kp7);
        kp8str=num2str(kp8);

        comando=strcat('V,',kp5str,VG,kp6str,VG,kp2str,VG,kp1str,VG,kp7str,VG,
            kp8str,VG,kp4str,VG,kp3str);

        fprintf(fid,'%s\n',comando);

        cont1=cont1+1;
        cont2=cont2+1;
    end
    cont1=cont1+1;
    cont2=1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%>>--  SOMA OS VOLUMES  --<<:%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fprintf(fid,'%s\n','btol,1E-3');
fprintf(fid,'%s\n','VADD,1,2,3,4,5,6,7,8');
fprintf(fid,'%s\n','VADD,65,9,10,11,12,13,14,15,16');
fprintf(fid,'%s\n','VADD,1,17,18,19,20,21,22,23,24');
fprintf(fid,'%s\n','VADD,2,25,26,27,28,29,30,31,32');

```

```

fprintf(fid, '%s\n' , 'VADD,1,33,34,35,36,37,38,39,40');
fprintf(fid, '%s\n' , 'VADD,2,41,42,43,44,45,46,47,48');
fprintf(fid, '%s\n' , 'VADD,1,49,50,51,52,53,54,55,56');
fprintf(fid, '%s\n' , 'VADD,2,57,58,59,60,61,62,63,64');

```

```

#####:>>~~ CRIA A MALHA ~~<<:#####

```

```

fprintf(fid, '%s\n' , 'et,1,solid45');
fprintf(fid, '%s\n' , 'mshape,1,3d');
fprintf(fid, '%s\n' , 'mshkey,0');
fprintf(fid, '%s\n' , 'desize,1,1,1');
fprintf(fid, '%s\n' , 'vmesh,1');

```

```

#####:>>~~ ALTERA POSIÇÃO DOS NÓS -
~~<<:#####

```

```

fprintf(fid, '%s\n' , 'MODMSH,DETACH');
noinf=[80,81,87,92,97,102,107,112,116,33,85,90,95,100,105,110,2,1,34,3
6,131,158,185,212,239,4,3,38,37,132,159,186,213,240,266,292,45,44,136,
163,190,217,244,270,297,52,51,140,167,194,221,248,274,302,59,58,144,17
1,198,225,252,278,307,66,65,148,175,202,229,256,282,312,73,72,152,179,
206,233,260,286,317];

```

```

fprintf(fid, '%s\n'
, '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!');
raio=20;
conttheta=0;
Ni=8;
Nj=8;
contanos=1;

```

```

#####:>>~~ SUPERFÍCIE DE BÉZIER -
~~<<:#####

```

```

while raio<101

```

```

    while conttheta<9
        theta=conttheta*7.5;

```

```

        i=0;
        j=0;

```

```

        u=((raio-20)/80);
        v=theta/60;

```

```

        fprintf(fid, '%s\n' , 'bezier=0');

```

```

        while i<=Ni
            while j<=Nj
                aux1=factorial(Ni)/((factorial(i))*(factorial(Ni-i)));
                aux2=(u^i)*((1-u)^(Ni-i));
                aux3=factorial(Nj)/((factorial(j))*(factorial(Nj-j)));
                aux4=(v^j)*((1-v)^(Nj-j));
                blend=aux1*aux2*aux3*aux4;
                istr=num2str(i);
                jstr=num2str(j);
                blendstr=num2str(blend);
                comando=strcat('Pij=y',istr,jstr);
                fprintf(fid, '%s\n' , comando);
                comando=strcat('blendstr=',blendstr);

```

```

        fprintf(fid,'%s\n' ,comando);
        fprintf(fid,'%s\n' , 'bezier=bezier+((-Pij-
essess)*blendstr)');
        j=j+1;

    end
    j=0;
    i=i+1;
end

node=noinf(contanos);
nodestr=num2str(node);
raiostr=num2str(raio);
thetastr=num2str(theta);
bezier='bezier';

comando=strcat('mmodif,',nodestr,VG,raiostr,VG,thetastr,VG,bezier);
fprintf(fid,'%s\n' ,comando);

    fprintf(fid,'%s\n'
, '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!');

    contanos=contanos+1;
    conttheta=conttheta+1;
end

    conttheta=0;
    raio=raio+10;
end

```

```

#####:>>~- NÓS DO MEIO --<<:#####

```

```

nomid=[86,91,96,101,106,111,115,119,130,157,184,211,238,265,291,5,39,1
33,160,187,214,241,267,293,46,137,164,191,218,245,271,298,53,141,168,1
95,222,249,275,303,60,145,172,199,226,253,279,308,67,149,176,203,230,2
57,283,313,74,153,180,207,234,261,287,318];

```

```

fprintf(fid,'%s\n'
, '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!');
raio=25;
conttheta=1;
Ni=8;
Nj=8;
contanos=1;

```

```

#####:>>~- SUPERFÍCIE DE BÉZIER -
~<<:#####

```

```

while raio<101

    while conttheta<16
        theta=conttheta*3.75;

        i=0;
        j=0;

        u=((raio-20)/80);
        v=theta/60;
    end
end

```

```

fprintf(fid,'%s\n' , 'bezier=0');

while i<=Ni
    while j<=Nj
        aux1=factorial(Ni)/((factorial(i))*(factorial(Ni-i)));
        aux2=(u^i)*((1-u)^(Ni-i));
        aux3=factorial(Nj)/((factorial(j))*(factorial(Nj-j)));
        aux4=(v^j)*((1-v)^(Nj-j));
        blend=aux1*aux2*aux3*aux4;
        istr=num2str(i);
        jstr=num2str(j);
        blendstr=num2str(blend);
        comando=strcat('Pij=y',istr,jstr);
        fprintf(fid,'%s\n' ,comando);
        comando=strcat('blendstr=',blendstr);
        fprintf(fid,'%s\n' ,comando);
        fprintf(fid,'%s\n' , 'bezier=bezier+((-Pij-
essess)*blendstr)');
        j=j+1;
    end
    j=0;
    i=i+1;
end

node=nomid(contanos);
nodestr=num2str(node);
raiostr=num2str(raio);
thetastr=num2str(theta);
bezier='bezier';

comando=strcat('nmodif, ',nodestr,VG,raiostr,VG,thetastr,VG,bezier);
fprintf(fid,'%s\n' ,comando);

fprintf(fid,'%s\n'
, '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!');

contanos=contanos+1;
conttheta=conttheta+2;
end

conttheta=1;
raio=raio+10
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%:>>~- MULTIPLICA OS GOMOS -
~<<:%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%                                     Não usado

fprintf(fid,'%s\n' , 'vgen,3,1,,,120,,2000,,,');
fprintf(fid,'%s\n' , 'csys,0');
fprintf(fid,'%s\n' , 'vsym,x,1,,,6000,0,0');
fprintf(fid,'%s\n' , 'csys,1');
fprintf(fid,'%s\n' , 'vgen,2,4,,,60,,2000,,,');
fprintf(fid,'%s\n' , 'VDEL,4');
fprintf(fid,'%s\n' , 'vgen,3,5,,,120,,2000,,,');
fprintf(fid,'%s\n' , 'vadd,1,2,3,4,5,6');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%:>>~- CONSTRAINT EQUATIONS -
~<<:%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fprintf(fid,'%s\n' , 'CE,1,0,1,UX,0.866,1,UY,0.5,,,');

```

```

fprintf(fid,'%s\n','CE,2,0,3,UX,0.866,3,UY,0.5,,,');
fprintf(fid,'%s\n','CE,3,0,6,UX,0.866,6,UY,0.5,,,');
fprintf(fid,'%s\n','CE,4,0,7,UX,0.866,7,UY,0.5,,,');
fprintf(fid,'%s\n','CE,5,0,8,UX,0.866,8,UY,0.5,,,');
fprintf(fid,'%s\n','CE,6,0,116,UX,0.866,116,UY,0.5,,,');
fprintf(fid,'%s\n','CE,7,0,117,UX,0.866,117,UY,0.5,,,');
fprintf(fid,'%s\n','CE,8,0,295,UX,0.866,295,UY,0.5,,,');
fprintf(fid,'%s\n','CE,9,0,299,UX,0.866,299,UY,0.5,,,');
fprintf(fid,'%s\n','CE,10,0,292,UX,0.866,292,UY,0.5,,,');
fprintf(fid,'%s\n','CE,11,0,294,UX,0.866,294,UY,0.5,,,');
fprintf(fid,'%s\n','CE,12,0,297,UX,0.866,297,UY,0.5,,,');
fprintf(fid,'%s\n','CE,13,0,300,UX,0.866,300,UY,0.5,,,');
fprintf(fid,'%s\n','CE,14,0,302,UX,0.866,302,UY,0.5,,,');
fprintf(fid,'%s\n','CE,15,0,304,UX,0.866,304,UY,0.5,,,');
fprintf(fid,'%s\n','CE,16,0,305,UX,0.866,305,UY,0.5,,,');
fprintf(fid,'%s\n','CE,17,0,307,UX,0.866,307,UY,0.5,,,');
fprintf(fid,'%s\n','CE,18,0,309,UX,0.866,309,UY,0.5,,,');
fprintf(fid,'%s\n','CE,19,0,310,UX,0.866,310,UY,0.5,,,');
fprintf(fid,'%s\n','CE,20,0,312,UX,0.866,312,UY,0.5,,,');
fprintf(fid,'%s\n','CE,21,0,314,UX,0.866,314,UY,0.5,,,');
fprintf(fid,'%s\n','CE,22,0,315,UX,0.866,315,UY,0.5,,,');
fprintf(fid,'%s\n','CE,23,0,317,UX,0.866,317,UY,0.5,,,');
fprintf(fid,'%s\n','CE,24,0,319,UX,0.866,319,UY,0.5,,,');
fprintf(fid,'%s\n','CE,25,0,320,UX,0.866,320,UY,0.5,,,');
fprintf(fid,'%s\n','CE,26,0,120,UX,0.866,120,UY,0.5,,,');

```

```

fprintf(fid,'%s\n','CE,27,0,30,UY,1');
fprintf(fid,'%s\n','CE,28,0,31,UY,1');
fprintf(fid,'%s\n','CE,29,0,33,UY,1');
fprintf(fid,'%s\n','CE,30,0,34,UY,1');
fprintf(fid,'%s\n','CE,31,0,35,UY,1');
fprintf(fid,'%s\n','CE,32,0,38,UY,1');
fprintf(fid,'%s\n','CE,33,0,41,UY,1');
fprintf(fid,'%s\n','CE,34,0,43,UY,1');
fprintf(fid,'%s\n','CE,35,0,45,UY,1');
fprintf(fid,'%s\n','CE,36,0,48,UY,1');
fprintf(fid,'%s\n','CE,37,0,50,UY,1');
fprintf(fid,'%s\n','CE,38,0,52,UY,1');
fprintf(fid,'%s\n','CE,39,0,55,UY,1');
fprintf(fid,'%s\n','CE,40,0,57,UY,1');
fprintf(fid,'%s\n','CE,41,0,59,UY,1');
fprintf(fid,'%s\n','CE,42,0,62,UY,1');
fprintf(fid,'%s\n','CE,43,0,64,UY,1');
fprintf(fid,'%s\n','CE,44,0,66,UY,1');
fprintf(fid,'%s\n','CE,45,0,69,UY,1');
fprintf(fid,'%s\n','CE,46,0,71,UY,1');
fprintf(fid,'%s\n','CE,47,0,73,UY,1');
fprintf(fid,'%s\n','CE,48,0,76,UY,1');
fprintf(fid,'%s\n','CE,49,0,78,UY,1');
fprintf(fid,'%s\n','CE,50,0,80,UY,1');
fprintf(fid,'%s\n','CE,51,0,83,UY,1');
fprintf(fid,'%s\n','CE,52,0,129,UY,1');

```

```
fprintf(fid,'%s\n','finish');
```

```

#####:>>~- OBTENÇÃO DA SOLUÇÃO -
~<<:#####

```

```

fprintf(fid,'%s\n','/SOLU');
fprintf(fid,'%s\n','D,73,UZ,0');
fprintf(fid,'%s\n','D,73,UX,0');
fprintf(fid,'%s\n','F,80,FZ,-forca');
fprintf(fid,'%s\n','Solve');
fprintf(fid,'%s\n','finish');

```



```
#####>>~- COMANDOS POST --<<#####
```

```
fprintf(fid, '%s\n' , '/post1 ');
fprintf(fid, '%s\n' , 'set,last');
fprintf(fid, '%s\n' , 'obje=0');
```

```
nosup=[83,82,88,93,98,103,108,113,117,30,27,24,21,18,15,12,9,6,31,28,2
5,22,19,16,13,10,7,41,40,134,161,188,215,242,268,294,48,47,138,165,192
,219,246,272,299,55,54,142,169,196,223,250,276,304,62,61,146,173,200,2
27,254,280,309,69,68,150,177,204,231,258,284,314,76,75,154,181,208,235
,262,288,319];
```

```
fprintf(fid, '%s\n'
, '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!');
raio=20;
conttheta=0;
fprintf(fid, '%s\n' , 'csys,0');
contanos=1;
```

```
while raio<101
    while conttheta<9
        theta=conttheta*7.5;

        no=nosup(contanos);
        nostr=num2str(no);
        raiostr=num2str(raio);
        ux='u,x';
        uy='u,y';
        uz='u,z';
        maisnoux='+(noux*';
        maisnouy='+(nouy*';
        comando=strcat(' *get,noux,node, ',nostr,VG,ux);
        fprintf(fid, '%s\n' , comando);
        comando=strcat(' *get,nouy,node, ',nostr,VG,uy);
        fprintf(fid, '%s\n' , comando);
        comando=strcat(' *get,nouz,node, ',nostr,VG,uz);
        fprintf(fid, '%s\n' , comando);
        thetarad=theta*pi/180;
        coss=cos(thetarad);
        cosstr=num2str(coss);
        seno=sin(thetarad);
        senstr=num2str(seno);
        aux='')';

        comando=strcat('raio=',raiostr,maisnoux,cosstr,aux,maisnouy,senstr,
aux);
        fprintf(fid, '%s\n' , comando);
        fprintf(fid, '%s\n' , 'Er=(( (Rc**2)-(R**2))**0.5)-(( (Rc**2)-
(raio**2))**0.5)+((R**2)/(4*foco)) ');
        fprintf(fid, '%s\n' , 'Pr=((raio**2)/(4*foco)) ');
        fprintf(fid, '%s\n' , 'aux=nouz+(Er-Pr) ');
        fprintf(fid, '%s\n' , 'obje=obje+aux**2 ');

        contanos=contanos+1;
        conttheta=conttheta+1;
    end
    conttheta=0;
    raio=raio+10;
end
```



```

fprintf(fid,'%s\n' , 'finish ');

fprintf(fid,'%s\n' , '/opt');
fprintf(fid,'%s\n' , 'lgwrite,tridim,lgw,,remove');
fprintf(fid,'%s\n' , 'opanl,tridim,lgw');

fprintf(fid,'%s\n' , '!opvar,y00,dv,0,50,');
fprintf(fid,'%s\n' , '!opvar,y02,dv,0,50,');
fprintf(fid,'%s\n' , '!opvar,y03,dv,0,50,');
fprintf(fid,'%s\n' , '!opvar,y04,dv,0,50,');
fprintf(fid,'%s\n' , '!opvar,y05,dv,0,50,');
fprintf(fid,'%s\n' , '!opvar,y06,dv,0,50,');
fprintf(fid,'%s\n' , '!opvar,y07,dv,0,50,');

fprintf(fid,'%s\n' , 'opvar,y10,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y12,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y13,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y14,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y15,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y16,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y17,dv,0,50,');

fprintf(fid,'%s\n' , 'opvar,y20,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y22,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y23,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y24,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y25,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y26,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y27,dv,0,50,');

fprintf(fid,'%s\n' , 'opvar,y30,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y32,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y33,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y34,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y35,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y36,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y37,dv,0,50,');

fprintf(fid,'%s\n' , 'opvar,y40,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y42,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y43,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y44,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y45,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y46,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y47,dv,0,50,');

fprintf(fid,'%s\n' , 'opvar,y50,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y52,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y53,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y54,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y55,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y56,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y57,dv,0,50,');

fprintf(fid,'%s\n' , 'opvar,y60,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y62,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y63,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y64,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y65,dv,0,50,');
fprintf(fid,'%s\n' , 'opvar,y66,dv,0,50,');

```

```

fprintf(fid, '%s\n' , 'opvar,y67,dv,0,50,');

fprintf(fid, '%s\n' , 'opvar,y70,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y72,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y73,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y74,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y75,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y76,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y77,dv,0,50,');

fprintf(fid, '%s\n' , 'opvar,y80,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y82,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y83,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y84,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y85,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y86,dv,0,50,');
fprintf(fid, '%s\n' , 'opvar,y87,dv,0,50,');

fprintf(fid, '%s\n' , '!opvar,forca,dv,,100,');

fprintf(fid, '%s\n' , 'opvar,obje,obj,,,10E-20');

fprintf(fid, '%s\n' , 'opdata,tridim,opt');
fprintf(fid, '%s\n' , 'optype,first');
fprintf(fid, '%s\n' , 'opfirst,5');
fprintf(fid, '%s\n' , 'opexe');
fprintf(fid, '%s\n' , 'finish');

fclose(fid);

clear all;

```

Anexo 6: Lista de fabricantes cujos telescópios foram adicionados no gráfico de Custos *versus* Abertura

Astro-Physics
Celestron
Coulter
D&G Optical
Dark Star
Edmund
Jim's Mobile Inc.
Mag One
Meade
Morning Star
North Star
Obsession
Optical Guidance Systems
Orion
Parallax
Sky Designs
Sky Valley
Stargazer Steve
StarMaster
Starsplitter
Takahashi
Tectron
Tele Vue

Crédito de Imagens

A figura Nº 3, **Padrões de difração**, foi gerada através do software Aberrator, versão 2.52, um freeware de autoria de Cor Berrevoets Ritthem

A figura Nº 6, **Exemplo de teste de Foucault**, foi retirada da página 197 do seguinte livro:

TEXEREAU, J. **How to make a telescope**, second edition, Richmond: Willman-Bell, 1984, 429p.

Todas as outras imagens, do autor ou retiradas da internet

Bibliografia

HARRINGTON, P. Star Ware, second edition . Estados Unidos: John Wiley & Sons, 1998, 375p.

TEXEREAU, J. How to make a telescope, second edition, Richmond: Willman-Bell, 1984, 429p.

INGALLS, A.G. Amateur telescope making, second edition, Richmond: Willman-Bell, 1966, 568p.

Kelley, W.T. **Three innovations in a dobsonian**, Sky and Telescope, junho de 1992, p. 84

Adler, A. **Flexing spheres into high quality telescope mirrors**, Sky and telescope, novembro de 2000, p. 131

BOBARU, F.;MUKHERJEE, S. Meshless approach to shape optimization of linear thermoelastic solids, 15 de novembro de 2000

HOU, G.; SHEEN, j.; CHUANG, C. Shape-Sensitivity analysis and design optimization of linear, thermoelastic solids, AIAA Journal, Vol. 30, Nº2, Fevereiro de 1992

AGRAWAL, B.; TREANOR, K. Shape control of a beam using piezoelectric actuators, Smart Mater. Struct. 8, 1999

NAKAMURA, S. Numerical analysis and graphic visualization with Matlab, Prentice Hall, 1996, 477p.

Amateur Telescope Makers Archives, <http://astro.umsystem.edu/atm/>

PMC-5009, Otimização estrutural aplicada à engenharia mecânica, notas de aula.

Ansys, versão 5.0 Documentação de referência