

Projeto Ótimo de Treliças: Otimização Estrutural

Francisco A. M. Gomes Neto, Tadao Ando Jr.,

Departamento de matemática aplicada, IMECC/UNICAMP

13081-970, Campinas, SP

E-mail: chico@ime.unicamp.br, tadao@ime.unicamp.br,

1 Introdução

Neste trabalho procuramos determinar o projeto ótimo de uma treliça sujeita a cargas de forças, obedecendo restrições de material e de domínio para obter a estrutura mais rígida. Como descrito em [1], diferentes aspectos podem ser considerados ao projetar uma estrutura. Esses aspectos em conjunto com características físicas desejadas na estrutura final (e.g. a minimização de uma grandeza física e quantidades limitadas de material) dão origem a problemas de otimização, dentre os quais citamos os de dimensionamento, forma e topologia.

Na sua forma mais básica, o problema de projeto estrutural pede que determinemos se cada ponto do espaço contem ou não material. Uma maneira de responder essa pergunta é discretizar o domínio em questão e analisar cada elemento da malha. Podemos então enxergar as treliças como uma discretização de elementos finitos do problema de projeto estrutural geral.

As seções subseqüentes estão divididas da seguinte forma: na seção 2 descrevemos a abordagem escolhida para modelar o problema que nos leva à formulação escolhida, descrita sucintamente na seção 3; resolvemos então esse problema adaptando um algoritmo do tipo restauração inexata, descrito em linhas gerais na seção 4; especializações e alguns resultados numéricos são mostrados nas seções 5 e 6 respectivamente; a seção 7 descreve a continuação do trabalho e objetivos futuros.

2 Estrutura Base

Para modelar o problema, usamos a abordagem conhecida como estrutura base (veja exemplo na figura 1), na qual enxergamos um determinado subconjunto de conexões entre um conjunto fixo de nós como elementos em potencial da estrutura final, o que nos fornece um problema de otimização discreta. Neste cenário, treliças são convenientes pois permitem contornar as dificuldades do problema de programação inteira inerente a essa abordagem considerando como variáveis de projeto as seções transversais das barras (variáveis contínuas) e permitindo, para representar a ausência de uma barra, seções nulas. Resolvemos assim, simultaneamente,

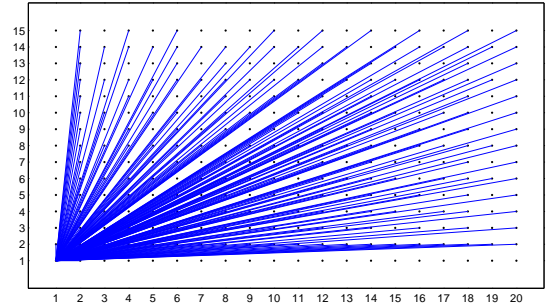


Figura 1: Exemplo de estrutura base para uma malha de 20 por 15 nós na qual todos estão conectados entre si por uma barra (nível de conexão máximo). Para manter a clareza da figura somente as barras que partem do nó inferior esquerdo foram representadas.

os problemas de dimensionamento e topologia.

Neste trabalho consideramos domínios retangulares e definimos o universo de barras da estrutura base escolhendo um nível de conexão desejado para a estrutura. Não há restrições para a posição inicial dos nós, entretanto a forma do domínio não entra no problema de otimização. Deste modo, é conveniente utilizar um grande número de nós para reduzir o efeito negativo de desconsiderar essa flexibilidade adicional. Essa escolha nos leva então a considerar opções para resolver problemas de grande porte, entre elas escolher modelos mais simples, reduzindo o número de variáveis, e especializar o algoritmo.

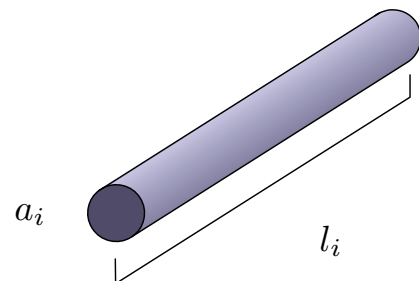


Figura 2: A i -ésima barra com volume $t_i = a_i l_i$. Assumimos que só há deformação em l_i e que a barra é composta de material linearmente elástico.

3 Formulação

Nos problemas descritos a seguir consideramos estruturas base com n nós, N graus de liberdade e m barras. Podemos ter dois ou três graus de liberdade possíveis por nó se considerarmos estruturas planas ou espaciais respectivamente. É necessário ainda restringir o movimento de alguns nós para prover apoios para a estrutura. Para uma treliça bidimensional temos $N < 2n$ e tipicamente $m = O(n^2)$. São dados do problema a carga de forças $f \in \mathbb{R}^N$, o volume total de material disponível $V \in \mathbb{R}, V > 0$ e a estrutura base. Usamos como variáveis de estado (descrevem a situação transitória da estrutura sob ação da carga) os deslocamentos nodais $u \in \mathbb{R}^N$ e como variáveis de projeto os volumes das barras representados por $t \in \mathbb{R}^m$ definidos à partir das grandezas representadas na figura 2. Finalmente, à cada barra corresponde ainda uma matriz de rigidez K_i que aparece nas restrições de equilíbrio dos problemas seguintes.

O trabalho é uma grandeza diretamente proporcional ao deslocamento que uma força causa em um corpo, assim podemos concluir que quanto mais rígida for uma estrutura menor o trabalho que uma carga é capaz de realizar sobre ela. O número $f^T u$ mede o trabalho total realizado por uma força externa f sobre uma treliça. Formulamos então o problema de determinar a treliça mais rígida para suportar uma carga f , com um volume de material V e com barras restritas a um subconjunto das barras que compõe a estrutura base como

$$\begin{aligned} \min_{u \in \mathbb{R}^N, t \in \mathbb{R}^m} \quad & f^T u \\ \text{sujeito à} \quad & \left(\sum_{i=1}^m t_i K_i \right) u = f \\ & \sum_{i=1}^m t_i = V \\ & t_i \geq 0, \quad i = 1 \dots m \end{aligned} \quad (1)$$

Demonstra-se em [2] que o problema 1 é equivalente à formulação na energia potencial

$$\begin{aligned} \max_{t \in \mathbb{R}^m} \min_{u \in \mathbb{R}^N} \quad & \left\{ \frac{1}{2} u^T \left(\sum_{i=1}^m t_i K_i \right) u - f^T u \right\} \\ & \sum_{i=1}^m t_i = V \\ & t_i \geq 0, \quad i = 1 \dots m \end{aligned} \quad (2)$$

Seguindo o caminho proposto em [1], trocamos a posição dos operadores min e max no problema 2 e introduzimos uma nova variável para chegar à uma formulação somente nos deslocamentos nodais,

suave e convexa

$$\begin{aligned} \min_{u \in \mathbb{R}^N, \alpha \in \mathbb{R}} \quad & \alpha - f^T u \\ \text{sujeito à} \quad & \frac{V}{2} [u^T K_i u] \leq \alpha \\ & i = 1, \dots, m \end{aligned} \quad (3)$$

com $N + 1$ variáveis e m restrições.

Para o caso de um único carregamento, os problemas acima podem ser reformulados como problemas de programação linear como descrito em [1]. Nosso objetivo é, entretanto, abordar o caso de múltiplos carregamentos, para o qual tais reformulações não permanecem válidas.

Podemos escrever esta formulação mais geral separando os vetores de deslocamento para cada uma das cargas e compondo, através de uma média ponderada, uma medida do trabalho total realizados pelas forças externas. Dados M carregamentos $f^k, k = 1, \dots, M$ e pesos $w_k, k = 1, \dots, M$ a formulação 3 pode ser reescrita como

$$\begin{aligned} \min_{u_p \in \mathbb{R}^N, \alpha \in \mathbb{R}} \quad & \alpha - \sum_{p=1}^M w_p f_p^T u_p \\ \text{sujeito à} \quad & \frac{V}{2} \sum_{p=1}^M [w_p u_p^T K_i u_p] \leq \alpha \\ & i = 1, \dots, m \\ & \sum_{p=1}^M w_p = 1 \end{aligned} \quad (4)$$

4 Restauração Inexata

No modelo descrito pelo problema 1 as restrições foram obtidas considerando as leis físicas que devem ser satisfeitas, salvo simplificações, para que uma treliça obtida como solução seja capaz de manter-se estável sob ação de uma carga fornecida. Na função objetivo associamos um custo, representado aqui pelo trabalho realizado pelas forças externas, à cada estado da estrutura. Sendo assim, fica evidente que o problema beneficia-se de métodos que privilegiem satisfazer as restrições em detrimento da obtenção do melhor valor do objetivo, pois uma estrutura que satisfaça as restrições de equilíbrio mas não seja a mais rígida ainda pode ser construída e desempenhar a função de transferir a carga para os apoios.

O método de Restauração Inexata descrito em [3] e referido de agora em diante como algoritmo IR, propõe uma alternativa para os métodos factíveis usuais ao enfatizar a factibilidade progressivamente com a proximidade da solução (mais especificamente, com a melhoria da aproximação da região factível no passo de minimização e com o ajuste do parâmetro de penalidade). Portanto, restrições

muito curvas não provocam encurtamento prematuro dos passos. Outra vantagem deste método é permitir o tratamento de restrições de desigualdade sem a introdução de variáveis de folga. No problema 3, que é o problema que desejamos resolver, a inclusão das folgas aumentaria o número de variáveis de $N + 1$ para $m + N + 1$.

Para descrever o método consideramos o problema de programação não linear genérico

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & g(x) \\ \text{sujeito à} \quad & C(x) \leq 0 \end{aligned} \quad (5)$$

O algoritmo IR, à semelhança dos procedimentos de programação quadrática seqüencial, divide cada iteração em duas fases: a primeira, de restauração, busca melhorar a factibilidade; a segunda, de minimização, melhora a otimalidade em relação à função objetivo.

4.1 Restauração

A fase de restauração é executada somente uma vez por iteração e obtém, à partir do iterando atual x^k , o ponto intermediário y^k que deve satisfazer às condições

$$\|C^+(y^k)\| \leq r \|C^+(x^k)\| \quad (6)$$

$$\|y^k - x^k\| \leq \beta \|C^+(x^k)\| \quad (7)$$

para $r \in [0, 1)$, $\beta > 0$ e $C^+(x) = \max\{C_j(x), 0\}$.

4.2 Minimização

A fase de minimização é executada após a fase de restauração e repetida até que o novo ponto $z^{k,i}$ satisfaça os critérios de aceitação baseados na função de mérito. Nesta etapa buscamos reduzir o valor do funcional f exigindo a manutenção da factibilidade até primeira ordem para algumas restrições e respeitando limites da região de confiança. O conjunto π_k é uma aproximação linear de uma seleção de restrições de $C(y^k)$ baseada em um parâmetro de tolerância p . Usar um subconjunto de restrições tem o intuito de diminuir o trabalho computacional por iteração

$$\begin{aligned} \pi_k = \{x \in \mathbb{R}^n \mid C_j(y^k) + C'_j(x - y^k) \leq C_j^+(y^k) \\ \text{quando } C_j(y^k) \geq -p\} \end{aligned} \quad (8)$$

O conjunto $B_{k,i}$ define a região de confiança centrada no ponto intermediário y^k

$$B_{k,i} = \{x \in \mathbb{R}^n \mid \|x - y^k\| \leq \delta_{k,i}\} \quad (9)$$

Assim, o subproblema da fase de minimização pode ser escrito como

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & g(x) \\ \text{sujeito à} \quad & x \in \pi_k \cap B_{k,i} \end{aligned} \quad (10)$$

cuja solução denominamos $z^{k,i}$.

4.3 Função de mérito

Métodos que trabalham com pontos infactíveis enfrentam o problema de julgar a qualidade dos iterandos levando em conta os objetivos conflitantes de reduzir a infactibilidade e o valor da função objetivo simultaneamente. Quando um ponto candidato melhora ou piora ambos temos uma noção intuitiva sobre a sua qualidade em relação ao ponto atual. Entretanto quando, por exemplo, o candidato ao próximo iterando melhora a factibilidade e piora a função objetivo, essa escolha não é clara. Uma maneira de realizar esse julgamento é usar as chamadas funções de mérito que são funcionais que combinam $g(x)$ e $C(x)$ para ordenar os pontos levando em consideração ambos os objetivos.

Usaremos a seguinte função de mérito

$$\psi(x, \theta) = \theta g(x) + (1 - \theta) \|C^+(x)\| \quad (11)$$

Para decidir a aceitação do ponto definimos a redução real, que é a diferença entre o valor da função de mérito no ponto final $z^{k,i}$ e no ponto atual x^k , e a redução prevista, que usa a mesma diferença mas desconsidera possíveis perdas na factibilidade causadas pela fase de minimização. Se a redução real A_{red} for maior ou igual a uma fração predefinida η da redução prevista P_{red} , aceitamos o ponto

$$A_{\text{red}} \geq \eta P_{\text{red}} \quad (12)$$

Entretanto se esta condição não é satisfeita, o ponto candidato é recusado, a região de confiança reduzida e a fase de minimização é executada novamente.

5 Especialização

Para o caso das treliças as matrizes de rigidez K_i podem ser escritas como produtos diádicos na forma

$$K_i = b_i b_i^T \quad (13)$$

onde b_i são vetores diretamente proporcionais às linhas da matriz de projeção entre o sistema de coordenadas das forças externas e os diversos sistemas de coordenadas alinhados com as barras ¹. Se considerarmos ainda o volume normalizado $V = 1$, o problema 3 pode ser reescrito como

¹Essa matriz de projeção é chamada de matriz de compatibilidade. As constantes de proporcionalidade envolvem o comprimento da barra l_i e o módulo de Young E_i

$$\begin{aligned} \min_{u \in \mathbb{R}^N, \alpha \in \mathbb{R}} \quad & \alpha - f^T u \\ \text{sujeito à} \quad & \frac{1}{2}(b_i^T u)^2 \leq \alpha \\ & i = 1, \dots, m \end{aligned} \quad (14)$$

O problema 14 é um caso especial do modelo geral mostrado no problema 5, com $x = [u, \alpha]^T$ e

$$\begin{aligned} g(x) &= g(u, \alpha) = \alpha - f^T u \\ C_i(x) &= C_i(u, \alpha) = \frac{1}{2}(b_i^T u)^2 - \alpha \end{aligned} \quad (15)$$

Definimos inicialmente a fase de restauração. Observando as equações 15 notamos que a variável artificial α é livre e aparece em todas as m restrições. Fazendo uso dessa propriedade propomos um ponto intermediário $\tilde{y}^k = [\tilde{y}_u^k, \tilde{y}_\alpha^k]^T$, sempre factível, obtido com um passo de restauração somente em α

$$\begin{aligned} \tilde{y}_u^k &= u^k \\ \tilde{y}_\alpha^k &= \max_i \frac{1}{2}(b_i^T u^k)^2 \end{aligned} \quad (16)$$

Entretanto o algoritmo IR pede que o ponto restaurado satisfaça as condições 6 e 7. Claramente é necessária uma escolha criteriosa de α e β ou uma alteração na atualização proposta para que o ponto seja aceito. É possível encontrar um ponto intermediário que satisfaça essas condições e que use um passo na variável α (sem mudar u) somente se escolhermos os parâmetros β e r satisfazendo

$$\beta + r \geq 1 \quad (17)$$

O passo proposto nas equações 16 satisfaz a condição 6 para qualquer valor de $r \in [0, 1)$ se $\beta \geq 1$. Entretanto se $\beta < 1$ o ponto \tilde{y}^k não vai satisfazer a condição 7. Neste caso, se $r < 1 - \beta$, não existirá passo que satisfaça as condições 6 e 7. Entretanto se $r \geq 1 - \beta$, definindo $y^k = [y_u^k, y_\alpha^k]^T$ conseguimos obter o passo reduzido

$$\begin{aligned} y_u^k &= \tilde{y}_u^k \\ y_\alpha^k &= \alpha + \beta(\tilde{y}_\alpha^k - \alpha) \end{aligned} \quad (18)$$

Iniciamos a análise da fase de minimização determinando a forma da região factível linearizada descrita na equação 8. Seja $z \in \pi_k$, com $z = [z_u, z_\alpha]^T$. Considerando as equações 8 e 15 concluímos que z deve satisfazer, para $i = 1, \dots, m$, a equação

$$(b_i^T y_u^k)(b_i^T z_u) - z_\alpha \leq \max \left\{ \xi - y_\alpha^k, \frac{1}{2}\xi \right\} \quad (19)$$

onde $\xi = (b_i^T y_u^k)^2$.

Assumimos ainda $p > \max |C(x^k)|$, i.e. todas as restrições são consideradas para formar a região π_k . O subproblema do passo de minimização fica portanto

$$\begin{aligned} \min_{z \in \mathbb{R}^{N+1}} \quad & z_\alpha - f^T z_u \\ \text{sujeito à} \quad & \text{condição (19)} \\ & \|z - y^k\| \leq \Delta_k \end{aligned} \quad (20)$$

6 Alguns resultados numéricos

Para obter os resultados descritos nesta seção implementamos o modelo mostrado no problema 3. Essa formulação foi escolhida por permitir uma série de extensões interessantes como múltiplos carregamentos, peso próprio da estrutura e reforço. A implementação da base do algoritmo bem como de um gerador automático de problemas foi feita em Matlab, escolhido pela facilidade de manipulação de dados. Entretanto enfrentamos um problema com a performance do código interpretado, contornado parcialmente pela transferência do núcleo de processamento do algoritmo, a fase de minimização, para uma biblioteca de ligação dinâmica compilada². Esse artifício nos permitiu resolver o subproblema 20 usando o pacote MINOS compilado em Fortran.

As figuras subseqüentes mostram alguns dos problemas resolvidos. Na legenda de cada figura são indicados o número de variáveis e o número de restrições. O número de variáveis é igual ao dobro do número de restrições (estruturas planas) mais um (variável artificial α) e o número de restrições é igual ao número de barras. Cada pequeno ponto representa uma rótula em potencial. Linhas pretas representam barras sofrendo compressão e linhas cinza representam barras sofrendo distensão. Os apoios são representados por quadrados quando o nó tem todos os graus de liberdade fixados e por triângulos apontando para a direita quando apenas a componente horizontal está fixa. As setas representam a carga de forças externas aplicada sobre a estrutura. Em todas os casos a estrutura base tem nível de conexão máximo, ou seja, todos os nós podem ser conectados por uma barra ou uma seqüência de barras colineares.

A figura 3 ilustra um modelo da vista lateral de uma ponte simples em dois níveis, com quatro pontos de apoio. Uma carga de forças é aplicada ao longo da linha 5 representando o peso da estrada.

No exemplo mostrado na figura 4 temos uma carga de forças similar que é aplicada na mesma linha dos apoios, entretanto neste caso restringimos o domínio de modo a permitir somente barras acima

²O Matlab chama essas bibliotecas de arquivos MEX e fornece uma API completa para transferência de dados com C ou Fortran

do nível da carga, resultando em uma estrutura familiar em forma de arco.

Na figura 5 ilustramos a possibilidade de reduzir a dimensão de problemas com simetria especular, como o mostrado na figura 4 (simétrico em relação ao eixo vertical que passa pela oitava coluna). Incluindo uma linha de apoios livres no sentido vertical obtemos uma boa aproximação para a metade esquerda da estrutura mostrada na figura 4.

As figuras 6 e 7 mostram uma extensão do artifício usado na obtenção da figura 5. Em ambos os casos apenas um apoio fixo é fornecido para suportar uma carga de forças que se estende pela largura da malha. Na primeira figura apoios deslizantes foram permitidos apenas na altura da linha 5. Na segunda figura destacamos que embora toda a extensão das laterais do domínio tenha sido oferecida como opção de apoio o projeto final faz uso de apenas dois.

As figuras 8 e 9 mostram duas configurações clássicas de solução conhecida, incluídas aqui com o intuito de validar a correção da implementação.

Finalmente, a figura 10 ilustra o fato de que a solução para o problema 3 pode ser um mecanismo³ em equilíbrio com as cargas de forças. A estrutura apresentada aqui claramente sofrerá uma rotação (que é um mecanismo) caso haja qualquer perturbação perpendicular à força f representada. Como na estrutura final temos várias componentes t_i nulas é natural que a matriz de rigidez global K definida por

$$K = \sum_{i=1}^m t_i K_i = \sum_{i=1}^m t_i (b_i b_i^T) \quad (21)$$

tenha posto incompleto. Notamos entretanto que esse mecanismo pode ser eliminado com a introdução de novas barras (e no caso específico deste exemplo de um apoio adicional). Ao introduzir novas barras, ou seja, tornar positivas algumas componentes t_i , estamos apenas aumentando de maneira seletiva o posto da matriz de rigidez. Assim, da mesma forma que podemos aumentar o posto de uma matriz adicionando à ela uma componente infinitesimal podemos eliminar um mecanismo adicionando à estrutura barras infinitesimais.

7 Conclusão

Neste trabalho descrevemos em linhas gerais o problema de treliças, a abordagem da estrutura base usada na modelagem e o algoritmo de Restauração Inexata proposto em [3]. Para resolver a reformulação mostrada na equação 3 propusemos especializações visando principalmente reduzir o custo computacional de cada iteração.

Para comprovar os resultados obtidos neste trabalho foi implementada também uma versão em

Matlab do algoritmo de pontos interiores descrito em [2]. Diversos problemas foram resolvidos usando ambos os métodos e seus resultados comparados quanto à disposição das barras e rigidez da estrutura. Estes resultados foram sempre muito próximos, raramente discordando na escolha dos elementos. Nos casos em que de fato foram diferentes, observamos que estas diferenças ficavam restritas à redução do volume de algumas barras (normalmente compensada com a introdução de outras mais finas) com pequena influência sobre a rigidez da estrutura final.

Como trabalho futuro pretendemos implementar o algoritmo descrito aqui em Fortran, pois optamos por não realizar comparações de desempenho usando as versões em Matlab e sim a implementação original⁴ em Fortran do algoritmo proposto em [2]. Entre outros objetivos, pretendemos ainda examinar métodos de identificação de restrições ativas para ajudar na escolha do parâmetro p mostrado na equação 8 e implementar o modelo da equação 4 que leva em conta múltiplos carregamentos.

Referências

- [1] M. P. Bendsøe & O. Sigmund, *Topology Optimization, Theory, Methods and Applications*, Springer, 4 (2003)
- [2] F. Jarre; M. Kočvara; J. Zowe, *Optimal truss design by interior point methods*, *SIAM Journal of Optimization* Vol. 8, No. 4, pp. 1084-1107, 1998
- [3] J. M. Martinez; E. A. Pilota, *Inexact - Restoration Algorithm for Constrained Optimization*, 1999

³Deslocamentos nodais que ocorrem sem a deformação da estrutura

⁴Gentilmente disponibilizada pelos autores de [2]

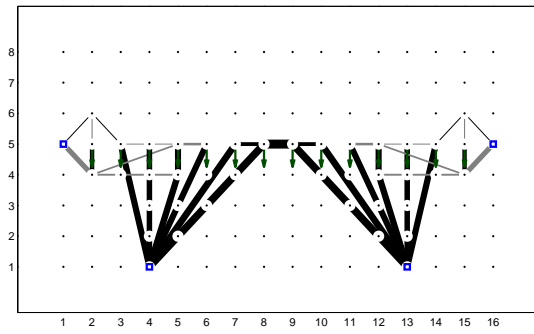


Figura 3: 249 variáveis com 3960 restrições

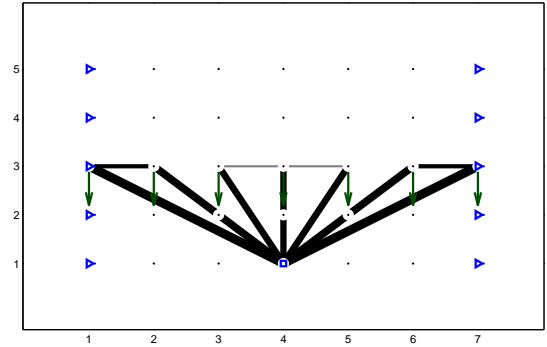


Figura 7: 59 variáveis com 386 restrições

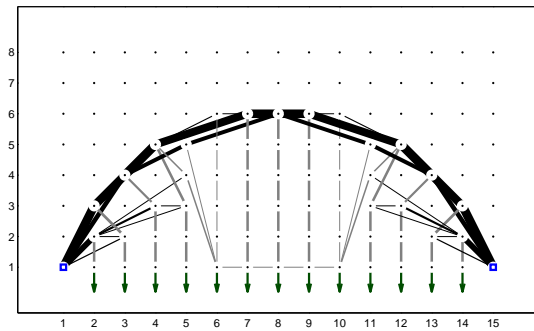


Figura 4: 237 variáveis com 3627 restrições

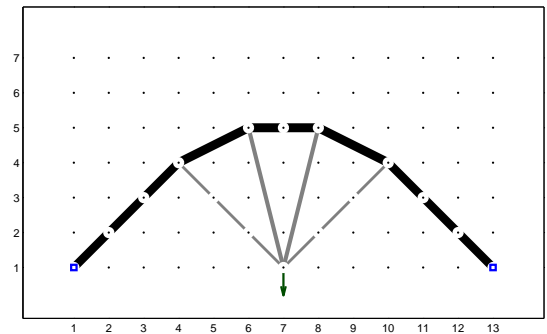


Figura 8: 179 variáveis com 2134 restrições

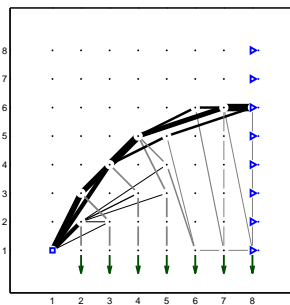


Figura 5: 119 variáveis com 1282 restrições

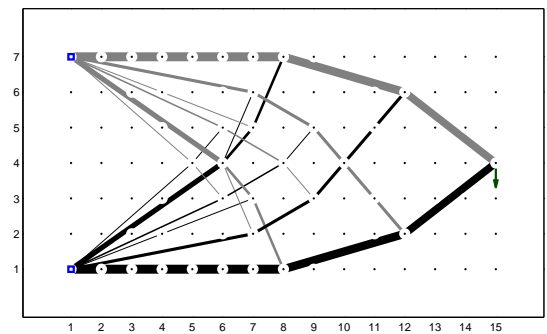


Figura 9: 207 variáveis com 3380 restrições

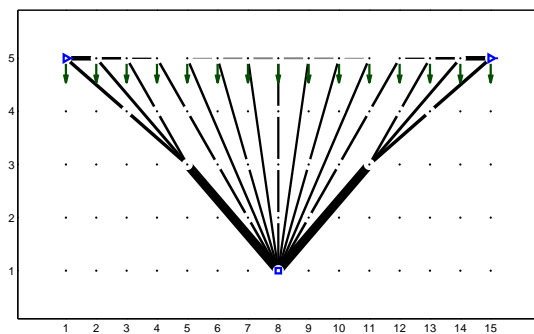


Figura 6: 147 variáveis com 1090 restrições

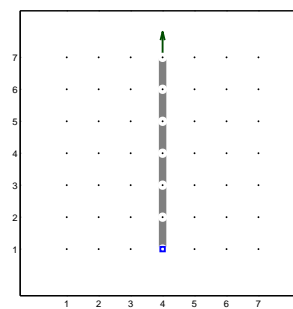


Figura 10: 97 variáveis com 748 restrições