
PROJETO 2

Arthur Assis Gonçalves

cc22300@g.unicamp.br

Vinícius Dos Santos Andrade

cc22333@g.unicamp.br

TI327 - Tópicos em Inteligência Artificial

Prof. Dr. Guilherme Macedo

Colégio Técnico de Campinas - UNICAMP

Campinas, 19 de Junho de 2024

Sumário

1	Introdução	4
2	Descrição	5
2.1	Descrição 1º problema	5
2.2	Descrição 2º problema	5
2.3	Descrição 3º problema	6
2.4	Descrição 4º problema	6
2.5	Descrição 5º problema	6
2.6	Descrição 7º problema	7
3	Desenvolvimento	8
3.1	Código 1º problema	8
3.2	Código 2º problema	11
3.3	Código 3º problema	12
3.4	Código 4º problema	13
3.5	Código 5º problema	15
3.6	Código 7º problema	16
4	Resultados	17
4.1	Testes Computacionais	17
4.2	Experimento	18
4.2.1	Ferramentas e Justificativas	18
4.2.2	Objetivos	18
4.2.3	Parâmetros e Resultados	19

4.3	Solução 1º Problema	20
4.3.1	Análise do Gráfico 2D (Figura 1)	20
4.3.2	Análise do Gráfico 3D (Figura 2)	21
4.4	Solução 2º Problema	23
4.4.1	Análise do Gráfico 2D (Figura 3)	23
4.4.2	Análise do Gráfico 3D (Figura 4)	24
4.5	Solução 3º Problema	26
4.5.1	Análise do Gráfico 2D (Figura 5)	26
4.5.2	Análise do Gráfico 3D (Figura 6)	27
4.6	Solução 4º Problema	29
4.6.1	Análise do Gráfico 2D (Figura 7)	29
4.6.2	Análise do Gráfico 3D (Figura 8)	30
4.7	Solução 5º Problema	32
4.7.1	Análise dos Gráficos 2D	32
4.7.2	Análise do Gráfico 3D (Figura 12)	33
4.8	Solução 7º Problema	36
5	Referências Bibliográficas	37

1 Introdução

A otimização linear, é uma técnica matemática utilizada para maximizar ou minimizar uma função linear sujeita a um conjunto de restrições lineares. Este método encontra aplicações em diversas áreas como economia, engenharia, logística e pesquisa operacional, sendo essencial para a tomada de decisões eficientes e racionais.

Apesar de sua extensa aplicação, a otimização possui algumas limitações. Os problemas relacionados à modelagem, solução e interpretação dos resultados podem impactar significativamente a eficiência e a aplicabilidade dos métodos de otimização linear. Entre os desafios mais comuns estão a complexidade computacional de grandes modelos, a dificuldade em lidar com dados imprecisos ou incompletos e as limitações inerentes das suposições de linearidade.

As funções de otimização são descritas de forma genérica por:

$$\begin{aligned} &\text{minimizar } c^T x \\ &\text{sujeito a } Ax = b \\ &x \in \mathbb{R}^+ \end{aligned} \tag{1}$$

Neste relatório, apresentamos a implementação do método Simplex em Python para resolver problemas de otimização linear, seguido da aplicação desse método em vários problemas de teste.

2 Descrição

Nesta seção, apresentaremos os problemas propostos no Projeto 2 da disciplina TI327 - Tópicos em Inteligência Artificial, ministrada pelo Prof. Dr. Guilherme Macedo. Os respectivos códigos utilizados para cada um dos seis problemas serão detalhados na seção 2.2,

2.1 Descrição 1º problema

Minimizar: $5x_1 + x_2$

Sujeito a:

$$2x_1 + x_2 \geq 6,$$

$$x_1 + x_2 \geq 4,$$

$$x_1 + 5x_2 \geq 10,$$

$$x_1, x_2 \geq 0.$$

2.2 Descrição 2º problema

Minimizar: $2x_1 - 3x_2$

Sujeito a:

$$x_1 + 2x_2 \leq 6,$$

$$2x_1 - x_2 \leq 8,$$

$$x_1, x_2 \geq 0.$$

2.3 Descrição 3º problema

Minimizar: $15(x_1 + 2x_2) + 11(x_2 - x_3)$

Sujeito a:

$$3x_1 \geq x_1 + x_2 + x_3,$$

$$0 \leq x_j \leq 1, \quad j = 1, 2, 3.$$

2.4 Descrição 4º problema

Minimizar: $10(x_3 + x_4)$

Sujeito a:

$$\sum_{j=1}^4 x_j = 400,$$

$$x_j - 2x_{j+1} \geq 0, \quad j = 1, 2, 3,$$

$$x_j \geq 0, \quad j = 1, 2, 3.$$

2.5 Descrição 5º problema

Maximizar: $-5x_1 + 3(x_1 + x_3)$

Sujeito a:

$$x_j + 1 \leq x_{j+1}, \quad j = 1, 2,$$

$$\sum_{j=1}^3 x_j = 12,$$

$$x_j \geq 0, \quad j = 1, 2, 3.$$

2.6 Descrição 7º problema

Maximizar: $9x_1 + 5x_2$

Sujeito a:

$$\sin\left(\frac{k}{13}\right)x_1 + \cos\left(\frac{k}{13}\right)x_2 \leq 7, \quad k = 1, \dots, 13$$

$$x_1, x_2 \geq 0$$

3 Desenvolvimento

3.1 Código 1º problema

```
1 from scipy.optimize import linprog
2 import numpy as np
3
4 def problema_1():
5     c = [5, 1]
6     A = [[-2, -1],
7          [-1, -1],
8          [-1, -5]]
9     b = [-6, -4, -10]
10
11     x0_bounds = (0, None)
12     x1_bounds = (0, None)
13
14     res = linprog(
15         c,
16         A_ub=A,
17         b_ub=b,
18         bounds=[x0_bounds, x1_bounds],
19         method='highs',
20         options={'disp': False}
21     )
22
23     print(f'A solucao otima deste problema foi x* = ({res.x[0]:.0f}, {res.
24 x[1]:.0f}) com f(x*) = {res.fun:.0f}.')
25
26     plot_2d_problema_1(res)
27     plot_3d_problema_1(res)
28
29     return res
```


Este código foi desenvolvido para solucionar o primeiro problema proposto na seção 2.1. Adotamos essa abordagem para resolver todos os problemas apresentados, alterando apenas nos casos em que o objetivo era a maximização; para esses, os coeficientes da função objetivo foram invertidos.

O código inicia com a definição dos coeficientes da função objetivo que será minimizada, representada pela lista

$$c = [5, 1],$$

onde a função objetivo é dada por

$$f(x) = 5x_1 + x_2.$$

Em seguida, estabelece-se a matriz A , contendo os coeficientes das restrições de desigualdade, onde cada linha corresponde a uma restrição e cada coluna, a uma variável. As restrições definidas são

$$-2x_1 - x_2 \leq -6,$$

$$-x_1 - x_2 \leq -4,$$

$$-x_1 - 5x_2 \leq -10,$$

que podem ser reescritas como

$$2x_1 + x_2 \geq 6,$$

$$x_1 + x_2 \geq 4,$$

$$x_1 + 5x_2 \geq 10,$$

respectivamente.

O vetor

$$b = [-6, -4, -10]$$

contém os termos independentes das restrições de desigualdade. Os limites das variáveis são definidos pelos pares

$$x0_bounds = (0, None),$$

$$x1_bounds = (0, None),$$

garantindo que ambas as variáveis sejam não-negativas, ou seja,

$$x_1 \geq 0 \text{ e } x_2 \geq 0.$$

A resolução do problema de programação linear é realizada através da função `linprog`, utilizando o solver 'highs'. Os parâmetros fornecidos incluem os coeficientes da função objetivo, a matriz de restrições, os termos independentes, os limites das variáveis e o método de otimização escolhido. Após a resolução, a solução ótima é apresentada, onde os valores das variáveis x_1 e x_2 são exibidos juntamente com o valor da função objetivo $f(x)$.

Por fim, o código invoca funções para a plotagem de gráficos bidimensionais e tridimensionais que ilustram a solução ótima encontrada e retorna o objeto contendo os resultados da otimização, que inclui tanto a solução ótima quanto o valor da função objetivo.

3.2 Código 2º problema

```
1 from scipy.optimize import linprog
2 import numpy as np
3
4 def problema_2():
5     c = [-2, 3]
6     A = [[1, 2],
7          [2, -1]]
8     b = [6, 8]
9
10    x0_bounds = (0, None) # x1 >= 0
11    x1_bounds = (0, None) # x2 >= 0
12
13    res = linprog(
14        c,
15        A_ub=A,
16        b_ub=b,
17        bounds=[x0_bounds, x1_bounds],
18        method='highs',
19        options={'disp': False}
20    )
21
22    print(f'A solucao otima deste problema e x* = ({res.x[0]:.0f}, {res.x
23    [1]:.0f}) com f(x*) = {-res.fun:.0f}.')
24
25    plot_2d_problema_2(res)
26    plot_3d_problema_2(res)
27
28    return res
```

3.3 Código 3º problema

```
1 from scipy.optimize import linprog
2 import numpy as np
3
4 def problema_3():
5     c = [-15, -41, 11] # Original: 15x1 + 41x2 - 11x3
6     A = [[-2, -1, -1]]
7     b = [-3]
8
9     x_bounds = [(0, 1), (0, 1), (0, 1)]
10
11     res = linprog(
12         c,
13         A_ub=A,
14         b_ub=b,
15         bounds=x_bounds,
16         method='highs',
17         options={'disp': False}
18     )
19
20     print(f'A solucao otima deste problema e x* = ({res.x[0]:.0f}, {res.x
21 [1]:.0f}, {res.x[2]:.0f}) com f(x*) = {-res.fun:.0f}.')
22
23     plot_2d_problema_3(res)
24     plot_3d_problema_3(res)
25
26     return res
```

3.4 Código 4º problema

```
1 from scipy.optimize import linprog
2 import numpy as np
3
4 def problema_4():
5     c = [0, 0, 10, 10]
6     A_ub = [
7         [-1, 2, 0, 0],
8         [0, -1, 2, 0],
9         [0, 0, -1, 2]
10    ]
11    b_ub = [0, 0, 0]
12
13    A_eq = [[1, 1, 1, 1]]
14    b_eq = [400]
15
16    bounds = [(0, None), (0, None), (0, None), (0, None)]
17
18
19    res = linprog(
20        c,
21        A_ub=A_ub,
22        b_ub=b_ub,
23        A_eq=A_eq,
24        b_eq=b_eq,
25        bounds=bounds,
26        method='highs',
27        options={'disp': False}
28    )
29
30    print(f'A solucao otima deste problema e x* = ({res.x[0]:.0f}, {res.x
31    [1]:.0f}, {res.x[2]:.0f}, {res.x[3]:.0f}) com f(x*) = {res.fun:.0f}.')
32
33    plot_2d_problema_4(res)
34    plot_3d_problema_4(res)
```

34

35

```
return res
```

3.5 Código 5º problema

```
1 from scipy.optimize import linprog
2 import numpy as np
3 def problema_5():
4     c = [2, -3, -3]
5     A_ub = [
6         [1, -1, 0],
7         [0, 1, -1]
8     ]
9     b_ub = [-1, -1]
10    A_eq = [[1, 1, 1]]
11    b_eq = [12]
12
13    bounds = [(0, None), (0, None), (0, None)]
14
15    res = linprog(
16        c,
17        A_ub=A_ub,
18        b_ub=b_ub,
19        A_eq=A_eq,
20        b_eq=b_eq,
21        bounds=bounds,
22        method='highs'
23    )
24    if res.success:
25        x_opt = res.x
26        f_opt = -res.fun
27        print(f"A solucao otima deste problema e x* = ({x_opt[0]:.0f}, {
28            x_opt[1]:.0f}, {x_opt[2]:.0f}) com f(x*) = {f_opt:.2f}.")
29        plot_2d_problema_5(res)
30        plot_3d_problema_5(res)
31    else:
32        print("O problema de otimizacao nao encontrou uma solucao viavel.")
33    )
34    return res
```

3.6 Código 7º problema

```
1 from scipy.optimize import linprog
2 import numpy as np
3
4 def problema_7():
5     c = [-9, -5]
6     A_ub = np.array([[np.sin(k / 13), np.cos(k / 13)] for k in range(1,
7     14)])
8     b_ub = np.array([7] * 13)
9
10     bounds = [(0, None), (0, None)] # x1 >= 0, x2 >= 0
11
12     res = linprog(
13         c,
14         A_ub=A_ub,
15         b_ub=b_ub,
16         bounds=bounds,
17         method='highs',
18         options={'disp': False}
19     )
20
21     if res.success:
22         plot_2d_problema_7(res)
23         plot_3d_problema_7(res)
24
25         print(r'A solucao otima deste problema e $x* = ({:.2f}, {:.2f})$
26         com $f(x*) = {:.2f}$.'.format(res.x[0], res.x[1], -res.fun))
27     else:
28         print("Otimizacao falhou. Status:", res.status)
29         print("Mensagem:", res.message)
30
31     return res
```


4 Resultados

4.1 Testes Computacionais

Os testes foram realizados em dois dispositivos distintos:

- **Notebook**
 - **CPU:** Intel Core i5 12500H
 - **RAM:** 8 GB DDR5
 - **GPU:** NVIDIA GeForce RTX 3050 Laptop
- **Desktop**
 - **CPU:** AMD Ryzen 3 3300X
 - **RAM:** 2 x 8 GB DDR4
 - **GPU:** NVIDIA GeForce GTX 1660 SUPER

4.2 Experimento

Para a realização dos experimentos, utilizamos a IDE PyCharm Professional 2024.1.3 (build 241.17890.14) com a versão 3.12.3 do Python. A biblioteca central para a análise foi a SciPy 1.13.1, explorando a função `linprog(method='highs')` para a resolução eficiente dos problemas de programação linear.

4.2.1 Ferramentas e Justificativas

A escolha da função `linprog(method='highs')` se deve à sua capacidade de selecionar automaticamente o solver mais adequado entre o método dual simplex revisado (`'highs-ds'`) e o método de ponto interior (`'highs-ipm'`), ambos da biblioteca **HIGHS**. Essa abordagem híbrida garante robustez, precisão e desempenho na resolução de problemas de programação linear, mesmo em larga escala.

Além disso, a função `linprog` oferece uma interface intuitiva para definir a função objetivo, as restrições e os parâmetros do problema, facilitando a implementação e análise dos resultados.

4.2.2 Objetivos

Com este experimento, buscamos:

1. Resolver de forma eficiente e precisa os problemas de otimização linear propostos.
2. Validar a implementação do método Simplex.
3. Avaliar o desempenho da biblioteca SciPy na resolução de problemas de otimização.

4.2.3 Parâmetros e Resultados

Os parâmetros utilizados na função `linprog` foram:

- `c`: Vetor de coeficientes da função objetivo.
- `A_ub`: Matriz de coeficientes das restrições de desigualdade.
- `b_ub`: Vetor de termos independentes das restrições de desigualdade.
- `bounds`: Tupla definindo os limites das variáveis de decisão.
- `method`: Método de resolução ('highs').
- `options`: Opções adicionais, como exibição de mensagens ('disp').

A função `linprog` retorna um objeto `OptimizeResult` contendo:

- `x`: Valores ótimos das variáveis de decisão.
- `fun`: Valor ótimo da função objetivo.
- `success`: Indicador de sucesso na resolução.
- `status`: Código de status da otimização.
- `message`: Mensagem sobre o resultado.

4.3 Solução 1º Problema

As Figuras 1 e 2 ilustram a representação gráfica do Problema 1 em duas e três dimensões, respectivamente. Em ambas as figuras, buscamos visualizar a região factível, definida pelas restrições do problema, e a solução ótima que maximiza a função objetivo.

4.3.1 Análise do Gráfico 2D (Figura 1)

Região Factível: A área sombreada em cinza claro na Figura 1 representa a região factível do problema, ou seja, o conjunto de todos os pontos (x_1, x_2) que satisfazem simultaneamente todas as restrições.

Restrições: Cada restrição é representada por uma reta, e a região factível está sempre do lado "permitido" da desigualdade:

$$2x_1 + x_2 \leq 6$$

$$x_1 + x_2 \leq 4$$

$$x_1 + 5x_2 \geq 10$$

As áreas abaixo das duas primeiras retas e acima da terceira reta representam os pontos que satisfazem as respectivas restrições.

Solução Ótima: A estrela vermelha, localizada em $(x_1, x_2) = (0, 6)$, representa a solução ótima do problema. Este é o ponto dentro da região factível que maximiza a função objetivo.

4.3.2 Análise do Gráfico 3D (Figura 2)

A Figura 2 complementa a análise com uma perspectiva tridimensional.

Superfície da Função Objetivo: O plano inclinado representa a função objetivo $f(x_1, x_2)$. A inclinação do plano indica como a função objetivo varia em relação a x_1 e x_2 .

Projeção da Região Factível: A região factível, definida pelas mesmas restrições lineares do gráfico 2D, é projetada no plano x_1x_2 .

Conexão da Solução Ótima: A linha tracejada conecta o ponto da solução ótima no espaço 3D à sua projeção na região factível no plano x_1x_2 , facilitando a visualização da localização da solução ótima.

Em ambas as visualizações, a solução ótima é consistente:

$$x^* = (0, 6) \text{ com } f(x^*) = 6.$$

A representação 3D oferece uma visão adicional da relação entre a função objetivo e a região factível, mostrando como a função objetivo cresce à medida que nos aproximamos da solução ótima, ilustrando a natureza da otimização linear.

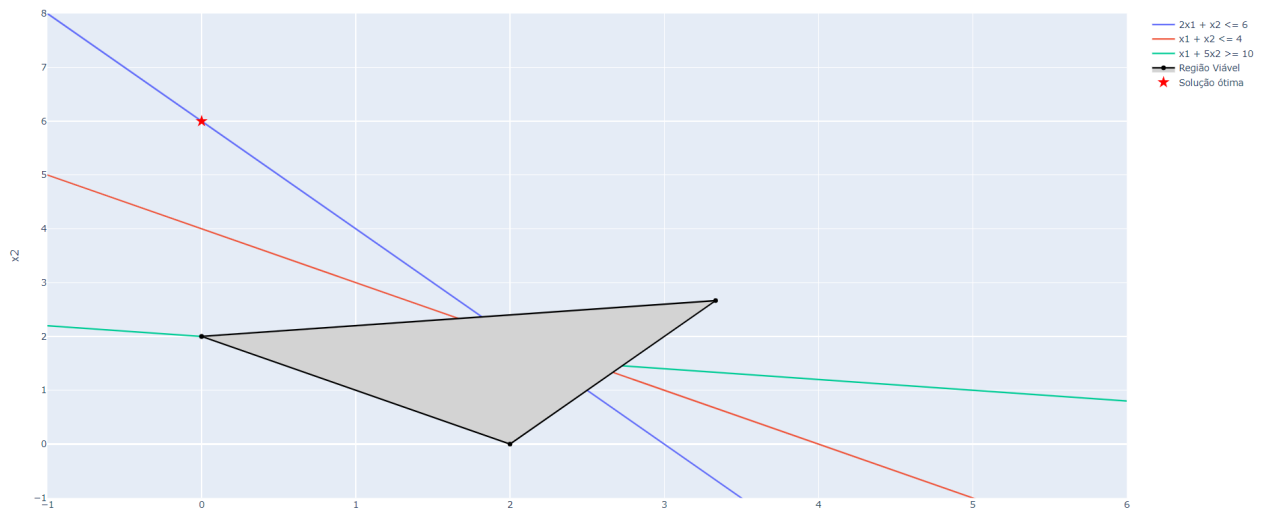


Figura 1: Plot 2D do Problema 1: Região factível (cinza claro) delimitada pelas restrições lineares, com a solução ótima destacada em vermelho.

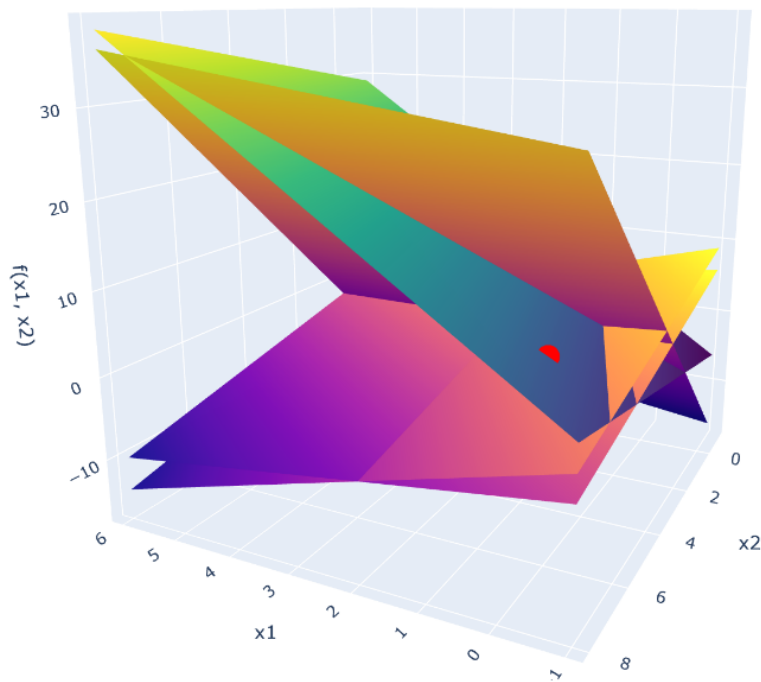


Figura 2: Plot 3D do Problema 1: Superfície da função objetivo (plano inclinado), região factível projetada no plano x_1x_2 , e solução ótima conectada ao plano x_1x_2 por uma linha tracejada.

4.4 Solução 2º Problema

As Figuras 3 e 4 ilustram a representação gráfica do Problema 2 em duas e três dimensões, respectivamente. Em ambas as figuras, buscamos visualizar a região factível, definida pelas restrições do problema, e a solução ótima que maximiza a função objetivo.

4.4.1 Análise do Gráfico 2D (Figura 3)

Região Factível: A área sombreada em cinza claro na Figura 3 representa a região factível do problema, ou seja, o conjunto de todos os pontos (x_1, x_2) que satisfazem simultaneamente todas as restrições.

Restrições: Cada restrição é representada por uma reta, e a região factível está sempre do lado "permitido" da desigualdade:

$$x_1 + 2x_2 \leq 6$$

$$2x_1 - x_2 \leq 8$$

As áreas abaixo de ambas as retas representam os pontos que satisfazem as respectivas restrições.

Solução Ótima: A estrela vermelha, localizada em $(x_1, x_2) = (4, 0)$, representa a solução ótima do problema. Este é o ponto dentro da região factível que maximiza a função objetivo.

4.4.2 Análise do Gráfico 3D (Figura 4)

A Figura 4 complementa a análise com uma perspectiva tridimensional.

Superfície da Função Objetivo: O plano inclinado representa a função objetivo $f(x_1, x_2)$. A inclinação do plano indica como a função objetivo varia em relação a x_1 e x_2 .

Projeção da Região Factível: A região factível, definida pelas mesmas restrições lineares do gráfico 2D, é projetada no plano x_1x_2 .

Conexão da Solução Ótima: A linha tracejada conecta o ponto da solução ótima no espaço 3D à sua projeção na região factível no plano x_1x_2 , facilitando a visualização da localização da solução ótima.

Em ambas as visualizações, a solução ótima é consistente:

$$x^* = (4, 0) \text{ com } f(x^*) = 8.$$

A representação 3D oferece uma visão adicional da relação entre a função objetivo e a região factível, mostrando como a função objetivo cresce à medida que nos aproximamos da solução ótima, ilustrando a natureza da otimização linear.

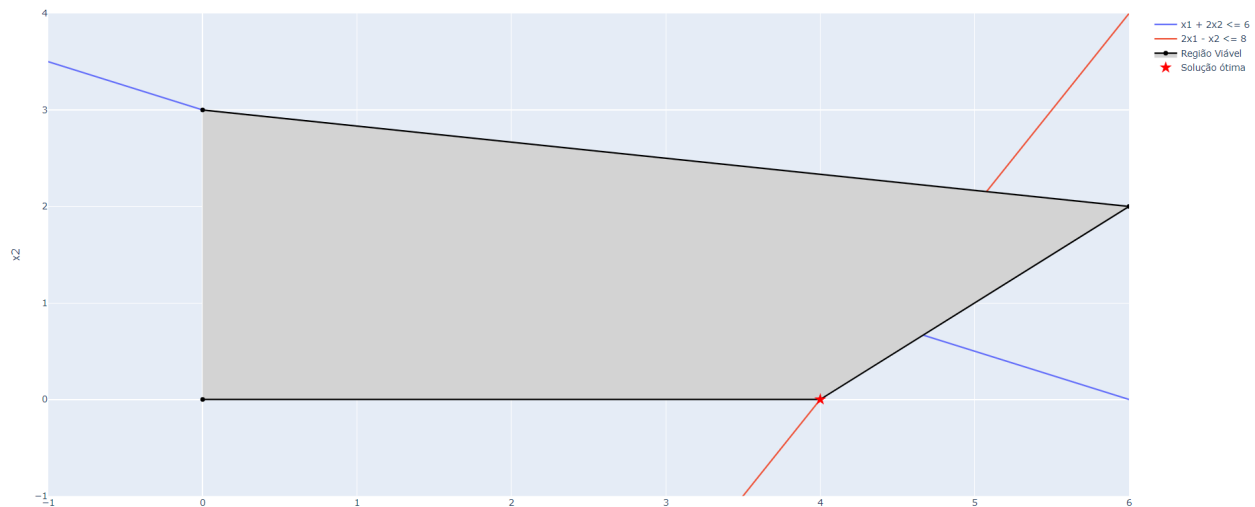


Figura 3: Plot 2D do Problema 2: Região factível (cinza claro) delimitada pelas restrições lineares, com a solução ótima destacada em vermelho.

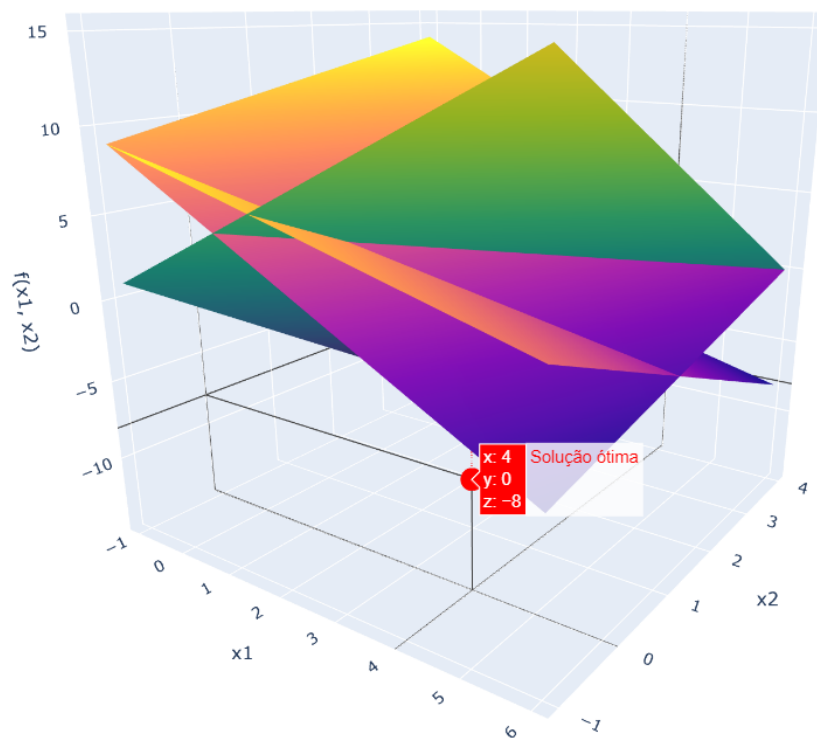


Figura 4: Plot 3D do Problema 2: Superfície da função objetivo (plano inclinado), região factível projetada no plano x_1x_2 , e solução ótima conectada ao plano x_1x_2 por uma linha tracejada.

4.5 Solução 3º Problema

As Figuras 5 e 6 ilustram a representação gráfica do Problema 3 em duas e três dimensões, respectivamente. Em ambas as figuras, buscamos visualizar a região factível, definida pelas restrições do problema, e a solução ótima que maximiza a função objetivo.

4.5.1 Análise do Gráfico 2D (Figura 5)

A Figura 5 apresenta uma seção transversal da região factível em duas dimensões, fixando $x_3 = 0$.

Região Factível: A área sombreada em cinza claro na Figura 5 representa a região factível do problema nesse plano, ou seja, o conjunto de todos os pontos (x_1, x_2) que satisfazem simultaneamente todas as restrições quando $x_3 = 0$.

Restrições: Cada restrição é representada por uma reta, e a região factível está sempre do lado "permitido" da desigualdade. As restrições, considerando $x_3 = 0$, são:

$$2x_1 - x_2 \geq 0$$

$$0 \leq x_1 \leq 1$$

$$0 \leq x_2 \leq 1$$

Solução Ótima: A estrela vermelha, localizada em $(x_1, x_2) = (1, 1)$, representa a solução ótima do problema projetada nesse plano.

4.5.2 Análise do Gráfico 3D (Figura 6)

A Figura 6 complementa a análise com uma perspectiva tridimensional da região factível.

Superfície da Função Objetivo: Os pontos roxos representam a função objetivo $f(x_1, x_2, x_3)$. A distribuição espacial desses pontos indica como a função objetivo varia em relação a x_1 , x_2 e x_3 .

Região Factível: A região factível, destacada em laranja, é definida pelas restrições do problema e delimita o espaço tridimensional onde a solução ótima pode estar localizada.

Conexão da Solução Ótima: A estrela vermelha, localizada em $(x_1, x_2, x_3) = (1, 1, 0)$, representa a solução ótima do problema. A linha tracejada conecta a solução ótima à sua projeção no plano x_1x_2 , facilitando a visualização da localização da solução.

Em ambas as visualizações, a solução ótima é consistente:

$$x^* = (1, 1, 0) \text{ com } f(x^*) = 56.$$

A representação 3D oferece uma visão mais completa da relação entre a função objetivo, as restrições e a solução ótima, ilustrando a natureza da otimização linear em um espaço tridimensional.

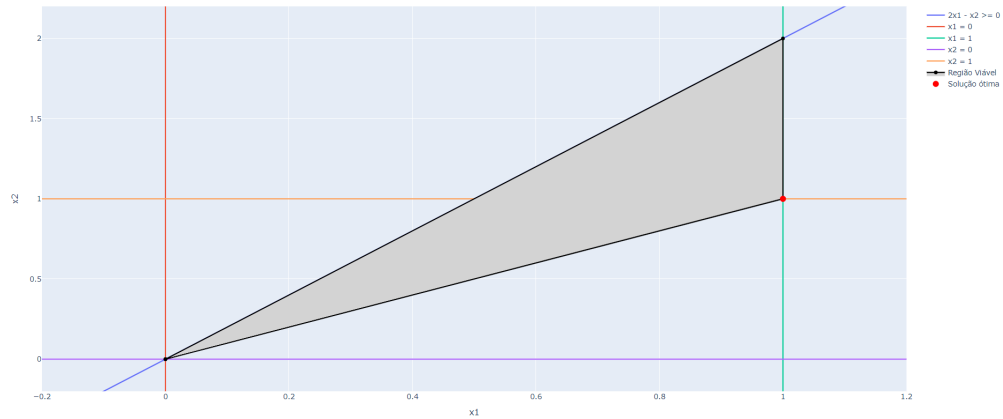


Figura 5: Plot 2D do Problema 3: Seção transversal da região factível com $x_3 = 0$. A região factível (cinza claro) é delimitada pelas restrições lineares, com a solução ótima projetada neste plano destacada em vermelho.

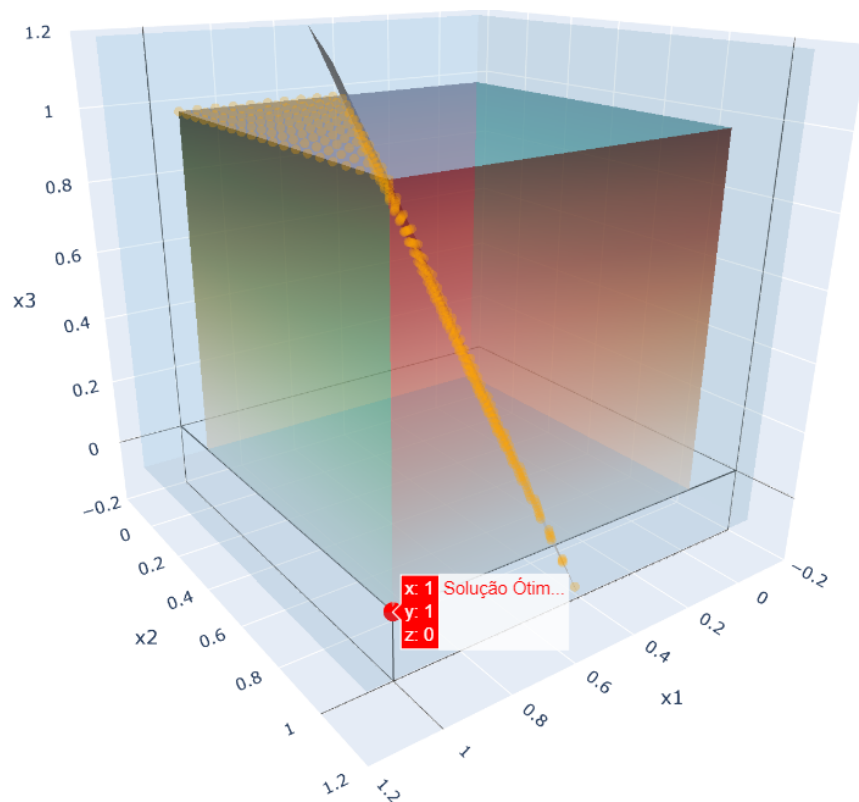


Figura 6: Plot 3D do Problema 3: Representação da função objetivo (pontos roxos), região factível (pontos laranjas) delimitada pelas restrições, e solução ótima (estrela vermelha) conectada ao plano x_1x_2 por uma linha tracejada.

4.6 Solução 4º Problema

As Figuras 7 e 8 ilustram a representação gráfica do Problema 4. Devido à alta dimensionalidade do problema (4 variáveis), a plotagem direta da região factível se torna complexa. Portanto, optamos por apresentar uma projeção 2D da região factível no plano x_1x_2 (Figura 7), fixando o valor de x_3 , e uma visualização 3D que destaca a solução ótima e as restrições projetadas em três dimensões (Figura 8).

4.6.1 Análise do Gráfico 2D (Figura 7)

A Figura 7 apresenta uma projeção da região factível no plano x_1x_2 , fixando x_3 em um valor específico.

Região Factível Projetada: A área sombreada em cinza claro na Figura 7 representa a região factível projetada, ou seja, o conjunto de todos os pontos (x_1, x_2) que, para o valor fixo de x_3 , satisfazem as restrições do problema.

Restrições Projetadas: Cada restrição é representada por uma reta, delimitando a região factível projetada. As restrições, considerando $x_4 = 400 - x_1 - x_2 - x_3$, são:

$$x_1 - 2x_2 \geq 0$$

$$x_2 - 2x_3 \geq 0$$

$$x_3 - 2x_4 \geq 0$$

$$x_1 + x_2 \leq 400 - x_3$$

Solução Ótima Projetada: A estrela vermelha na Figura 7 representa a solução ótima do problema projetada nesse plano, considerando o valor fixo de x_3 .

4.6.2 Análise do Gráfico 3D (Figura 8)

A Figura 8 complementa a análise com uma perspectiva tridimensional. Devido à dificuldade de representar um espaço de quatro dimensões, algumas restrições são apresentadas como planos para auxiliar a visualização.

Solução Ótima: A estrela vermelha na Figura 8 representa a solução ótima do problema em quatro dimensões, com a coordenada x_4 determinada pela restrição de igualdade ($x_4 = 400 - x_1 - x_2 - x_3$).

Planos de Restrições: Os planos representam as restrições do problema projetadas em três dimensões. A região factível, embora não diretamente visível devido à quarta dimensão, está contida na intersecção das regiões delimitadas por esses planos.

Em ambas as visualizações, a solução ótima, considerando todas as quatro variáveis, é:

$$x^* = (400, 0, 0, 0) \text{ com } f(x^*) = 0.$$

As representações gráfica 2D e 3D, apesar de limitadas pela alta dimensionalidade do problema, fornecem insights importantes sobre a solução do Problema 4.

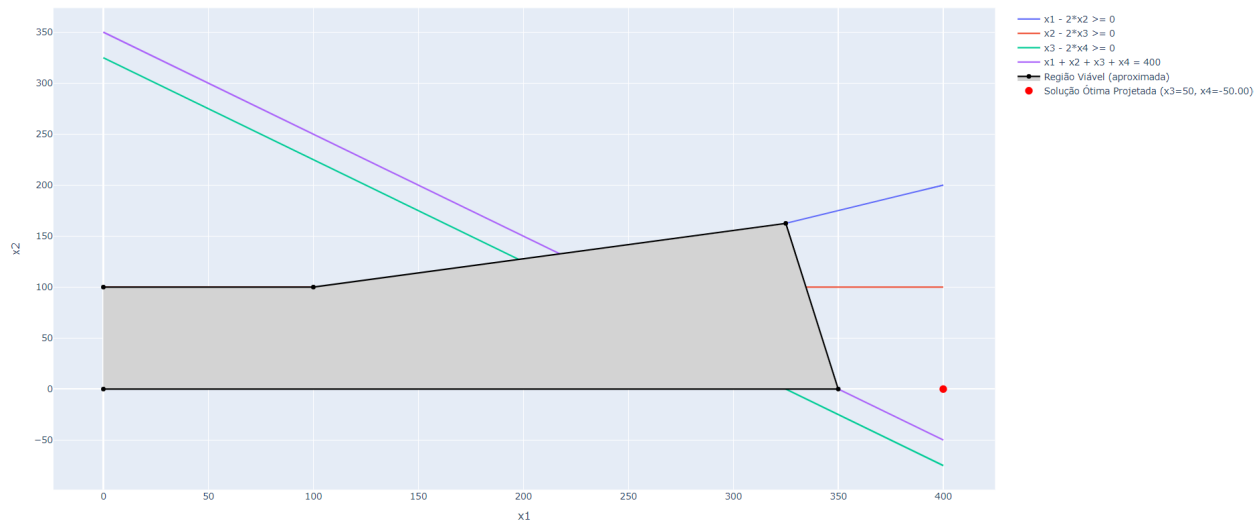


Figura 7: Plot 2D do Problema 4: Projeção da região factível no plano x_1x_2 , com x_3 fixo. A região viável projetada (cinza claro) é delimitada pelas restrições projetadas, com a solução ótima projetada neste plano destacada em vermelho.

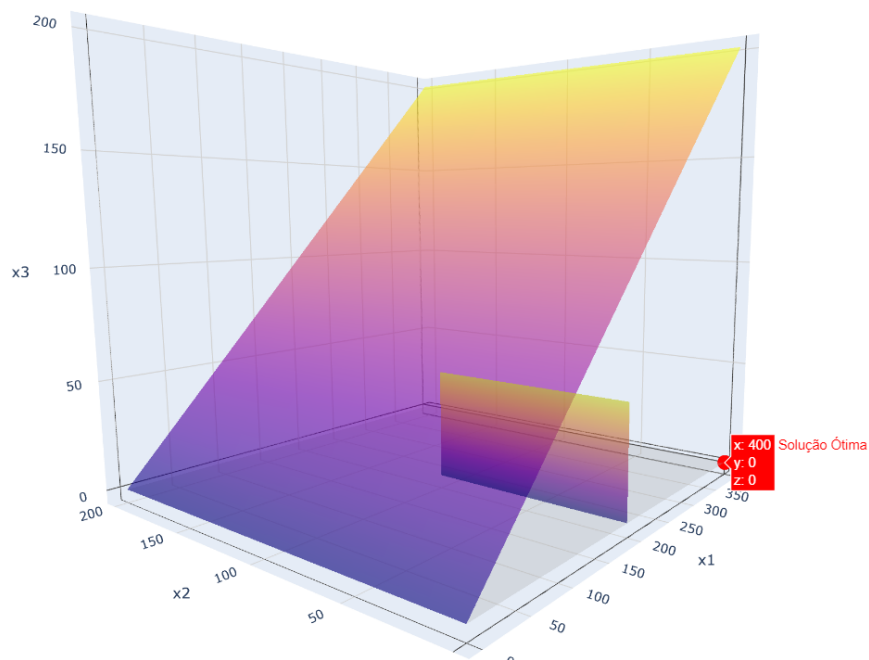


Figura 8: Plot 3D do Problema 4: Representação da solução ótima (estrela vermelha) e dos planos de restrições. Devido à alta dimensionalidade, a região factível não é diretamente visível, mas está contida na intersecção das regiões delimitadas pelos planos.

4.7 Solução 5º Problema

O Problema 5 apresenta um desafio interessante de visualização, pois a região factível é definida em um espaço tridimensional. Para facilitar a análise, empregamos duas abordagens:

1. Projeções 2D: As Figuras 9, 10 e 11 ilustram a região factível projetada em diferentes planos bidimensionais, fixando uma das variáveis de decisão por vez.

2. Visualização 3D: A Figura 12 apresenta uma representação tridimensional da região factível, exibindo a intersecção das restrições no espaço $x_1x_2x_3$.

4.7.1 Análise dos Gráficos 2D

Cada gráfico 2D apresenta uma projeção da região factível em um plano formado por duas variáveis, mantendo a terceira variável constante.

Figura 9 (x_3 fixo) A área sombreada em azul claro representa a região factível projetada no plano x_1x_2 , considerando um valor fixo para x_3 .

Figura 10 (x_2 fixo) A área sombreada em verde claro representa a região factível projetada no plano x_1x_3 , considerando um valor fixo para x_2 .

Figura 11 (x_1 fixo) A área sombreada em coral representa a região factível projetada no plano x_2x_3 , considerando um valor fixo para x_1 .

Em cada projeção, as restrições relevantes são representadas por retas, e a solução ótima projetada é destacada em vermelho.

4.7.2 Análise do Gráfico 3D (Figura 12)

A Figura 12 oferece uma visão espacial da região factível.

Planos de Restrições: Os planos coloridos representam as restrições de desigualdade e igualdade do problema. A região factível está contida na intersecção dos semi-espacos definidos por esses planos.

Solução Ótima: A esfera vermelha destaca a solução ótima do problema no espaço tridimensional.

Em todas as visualizações, a solução ótima, considerando as três variáveis, é:

$$x^* = (0, 1, 11) \text{ com } f(x^*) = 36.$$

A combinação de projeções 2D e visualização 3D facilita a compreensão da geometria do problema e da localização da solução ótima no espaço de decisão.

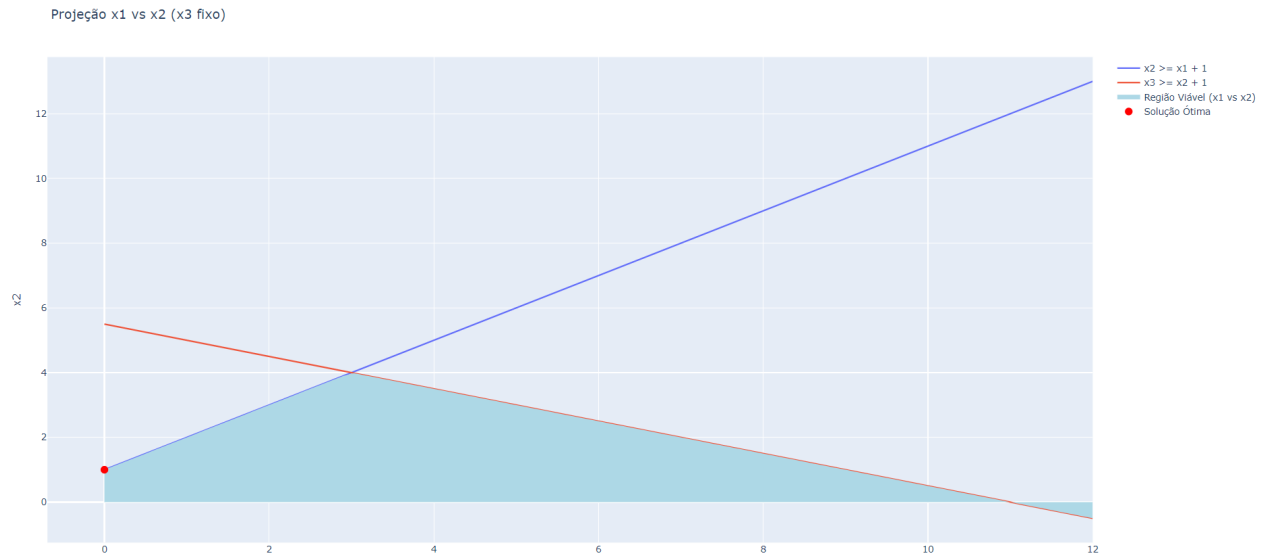


Figura 9: Projeção 2D do Problema 5: Região factível projetada no plano x_1x_2 com x_3 fixo. A região viável (azul claro) é delimitada pelas restrições projetadas, com a solução ótima projetada neste plano destacada em vermelho.

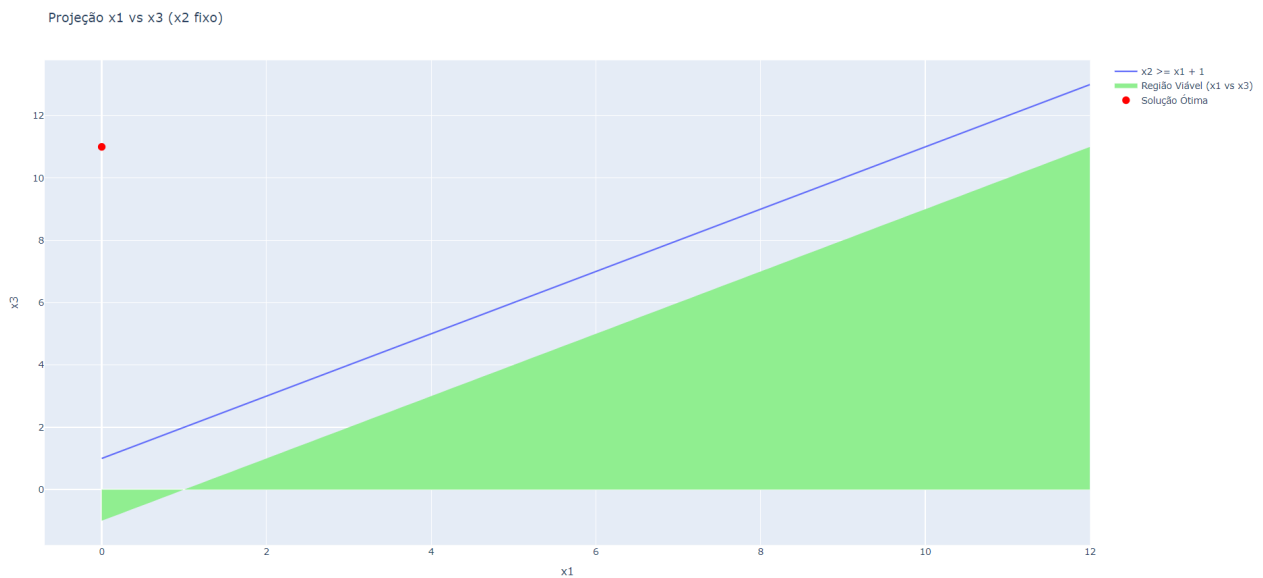


Figura 10: Projeção 2D do Problema 5: Região factível projetada no plano x_1x_3 com x_2 fixo. A região viável (verde claro) é delimitada pelas restrições projetadas, com a solução ótima projetada neste plano destacada em vermelho.

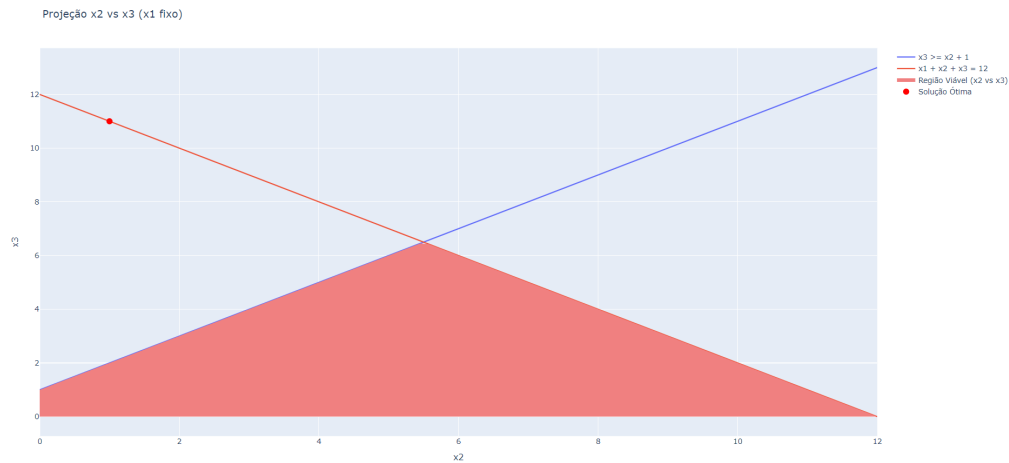


Figura 11: Projeção 2D do Problema 5: Região factível projetada no plano x_2x_3 com x_1 fixo. A região viável (coral) é delimitada pelas restrições projetadas, com a solução ótima projetada neste plano destacada em vermelho.

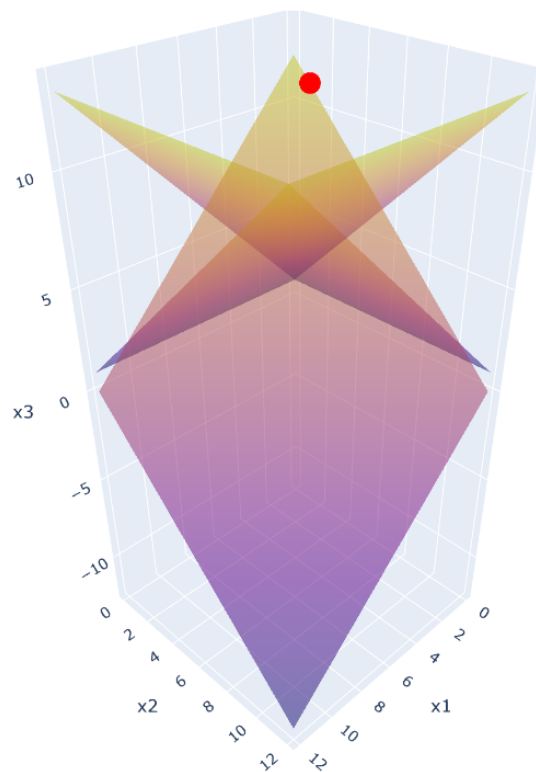


Figura 12: Visualização 3D do Problema 5: Representação da região factível delimitada pelos planos de restrições e da solução ótima (esfera vermelha).

4.8 Solução 7º Problema

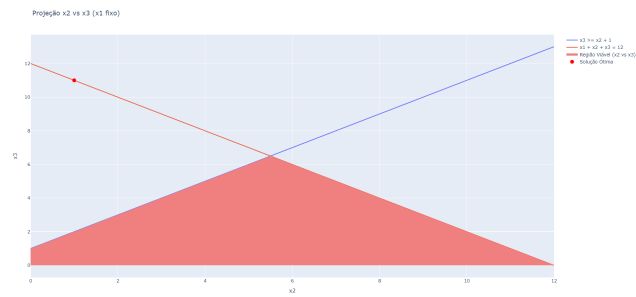


Figura 13: Plot 7

5 Referências Bibliográficas

- [1] Huangfu, Q., Galabova, I., Feldmeier, M., and Hall, J. A. J. "HiGHS - high performance software for linear optimization." <https://highs.dev/>
- [2] Huangfu, Q. and Hall, J. A. J. "Parallelizing the dual revised simplex method." *Mathematical Programming Computation*, 10 (1), 119-142, 2018. DOI: 10.1007/s12532-017-0130-5
- [3] Harris, Paula MJ. "Pivot selection methods of the Devex LP code." *Mathematical programming* 5.1 (1973): 1-28.
- [4] Goldfarb, Donald, and John Ker Reid. "A practicable steepest-edge simplex algorithm." *Mathematical Programming* 12.1 (1977): 361-371.
- [5] Virtanen, P. et al., 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17(3), pp.261-272.
- [6] SciPy documentation. <https://docs.scipy.org/doc/scipy/reference/>. Acessado em [data de acesso].
- [7] Dantzig, George B. *Linear programming and extensions*. Rand Corporation Research Study. Princeton Univ. Press, Princeton, NJ, 1963.
- [8] Hillier, S.H. and Lieberman, G.J. "Introduction to Mathematical Programming". McGraw-Hill, 1995. Chapter 4.
- [9] Bland, Robert G. "New finite pivoting rules for the simplex method." *Mathematics of Operations Research* 2 (1977): 103-107.
- [10] Andersen, Erling D., and Knud D. Andersen. "The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm." *High performance optimization*. Springer US, 2000. 197-232.

- [11] Andersen, Erling D. "Finding all linearly dependent rows in large-scale linear programming." *Optimization Methods and Software* 6.3 (1995): 219-227.
- [12] Freund, Robert M. "Primal-Dual Interior-Point Methods for Linear Programming based on Newton's Method." Unpublished Course Notes, March 2004. Accessed February 25, 2017. https://ocw.mit.edu/courses/sloan-school-of-management/15-084j-nonlinear-programming-spring-2004/lecture-notes/lec14_int_pt_mthd.pdf
- [13] Fourer, Robert. "Solving Linear Programs by Interior-Point Methods." Unpublished Course Notes, August 26, 2005. Accessed February 25, 2017. <http://www.4er.org/CourseNotes/Book%20B/B-III.pdf>
- [14] Andersen, Erling D., and Knud D. Andersen. "Presolving in linear programming." *Mathematical Programming* 71.2 (1995): 221-245.
- [15] Bertsimas, Dimitris, and J. Tsitsiklis. "Introduction to linear programming." *Athena Scientific* 1 (1997): 997.
- [16] Andersen, Erling D., et al. *Implementation of interior point methods for large scale linear programming*. HEC/Universite de Geneve, 1996.