

Otimização de processos químicos industriais: Avaliação da eficácia do *software* livre Scilab na resolução de problemas de programação não-linear com restrições

Chemical process optimization: Evaluation of effectiveness of open-source software Scilab in solve constrained nonlinear programming problems

Optimización de procesos químicos: Evaluación da eficacia do software livre Scilab para solución de problemas de programación non linear con restricciones

Recebido: 04/07/2021 | Revisado: 15/07/2021 | Aceito: 21/07/2021 | Publicado: 28/07/2021

Bruno Klier Silva

ORCID: <https://orcid.org/0000-0001-5678-1895>
Universidade Federal dos Vales do Jequitinhonha e Mucuri, Brasil
E-mail: brunoklier@outlook.com

Wu Hong Kwong

ORCID: <https://orcid.org/0000-0002-5258-1316>
Universidade Federal de São Carlos, Brasil
E-mail: wu@ufscar.br

Anamaria de Oliveira Cardoso

ORCID: <https://orcid.org/0000-0002-4518-075X>
Universidade Federal dos Vales do Jequitinhonha e Mucuri, Brasil
E-mail: Anamaria.cardoso@ict.ufvjm.edu.br

Resumo

A resolução de problemas de otimização não-linear com restrições de igualdade e desigualdade é uma tarefa difícil e tende a ser ainda mais difícil de ser resolvido quando envolve um grande número de variáveis. Sistemas de energia, controle e plantas industriais são representados por este tipo de problema e com o avanço tecnológico e computacional, é necessário ferramentas eficazes que consiga lidar com as características deste tipo de sistema, resolvendo de maneira eficaz o problema de otimização. Neste trabalho, dois problemas clássicos de otimização não-linear com restrições de processos químicos são resolvidos a partir do software livre Scilab, utilizando a função `fmincon`, de modo a avaliar o desempenho do algoritmo para a resolução de problemas de programação não-linear restrita. Os resultados obtidos foram comparados com as condições ótimas obtidas através do software Matlab. O software livre apresentou um bom desempenho na resolução dos problemas propostos, alcançando resultados eficazes, encontrando condições ótimas que resultaram em um aumento de cerca de 16% na função objetivo para o problema de maximização do retorno do investimento da planta de Williams-Otto, se mostrando uma alternativa eficaz para o desenvolvimento de pesquisas e tecnologias nesta área.

Palavras-chave: `Fmincon`; IpOpt; Planta de Williams-Otto; Otimização; Scilab.

Abstract

Solution of nonlinear optimization problems with equality and inequality constraints is a hard task and tend to be more complex when involves a large number of variables. Energy and control systems and chemical plants are represented by this type of problem. With technological and computational progress, its necessary use effective tools that can handle with characteristics of this type of systems, effectively solving optimization problem. In this work, we solve two classical problems of nonlinear optimization with constraints of chemical process in free software Scilab, using the function `fmincon`, in order to evaluate the performance of the algorithm for solving constrained nonlinear programming problems. Results are compared with optimal conditions obtained with Matlab. Free software shows a good performance, achieved effective results and finding points that results in a increase of 16% in objective function to investment return of Williams-Otto plant, showing itself an efficient alternative for the development of research and technology in this field.

Keywords: `Fmincon`; IpOpt; Williams-Otto Plant; Optimization; Scilab.

Resumen

Resolver problemas de optimización no lineal con restricciones de igualdad y desigualdad es una tarea difícil y tiende a ser aún más difícil cuando involucra una gran cantidad de variables. Los sistemas de potencia, control y plantas industriales están representados por este tipo de problemas y con los avances tecnológicos y computacionales se necesitan herramientas útiles que puedan abordar las características de este tipo de sistemas, resolviendo eficazmente el problema de optimización. En este trabajo, se resuelven dos problemas clásicos de optimización no lineal con restricciones de procesos químicos, utilizando el software libre Scilab, a través de una función fmincon disponible, con el fin de evaluar el desempeño del algoritmo para la resolución de problemas de programación no lineal. Los resultados obtenidos se compararon con las condiciones óptimas utilizando el software Matlab. El software libre presenta un buen desempeño en la resolución de los problemas propuestos, logrando resultados efectivos, encontrando condiciones óptimas que resultaron en un incremento de 16% en la función objetivo para el problema de maximizar el retorno de la inversión de la planta de Williams-Otto, si se muestra una alternativa eficaz para el desarrollo de investigaciones y tecnologías en esta área.

Palabras clave: Fmincon; IpOpt; Planta de Williams-Otto; Optimización; Scilab.

1. Introdução

A otimização é uma ferramenta matemática amplamente utilizada na indústria química, com diversas e mais diversas aplicações, desde as etapas de síntese, projeto e planejamento até a etapa de controle dos processos (Biegler, 2018; Ryoo & Sahinidis, 1995; Thakur et al., 2017). A solução de um problema de otimização consiste em encontrar os mínimos ou máximos de uma função de várias variáveis, com valores dentro de uma determinada região de espaço multi-dimensional (Martinez & Santos, 1995). Os problemas de otimização são classificados de acordo com o tipo de variáveis envolvidas (discretas ou contínuas) e a não-linearidade das funções e restrições que compõem estes problemas. Para problemas de processos químicos industriais comumente temos problemas de otimização que envolvem Programação Linear (LP), Programação Quadrática (QP), Programação Linear Inteiro Mista (MILP), Programação Não-Linear (NLP) e Programação Não-Linear Inteiro Mista (MINLP) (Chaudhary, 2009).

Quando se tem como foco sistema não-lineares, sendo essa a dinâmica de grande parte dos processos químicos atuais, a resolução dos problemas de otimização torna-se uma tarefa complexa, seja pela interação entre as unidades que compõem o sistema não pode ser negligenciada em virtude da alta integração dos processos através do ciclo de massa e energia, o que confere alto grau de não-linearidade a esse tipo de sistema, ou pelo alto número de variáveis e restrições envolvidas no problema de programação (Rangaiah & Kariwala, 2012).

De maneira geral, engenheiros possuem acesso a ferramentas eficazes na simulação e modelagem de processos industriais porém, muitas destas possuem limitações a quantidade de variáveis envolvidas no problema de otimização não-linear e ao tipo de restrições a que estes estão submetidos, limitando sua aplicação muitas vezes a problemas de otimização com grau de liberdade menor que 100, o que está longe de representar um problema real de forma satisfatória (Miller et al., 2018). Apesar disso, nas últimas décadas identifica-se um grande avanço no desenvolvimento de técnicas e ferramentas para resolução de problemas de programação não-linear, principalmente àquelas direcionadas aos sistemas de grandes dimensões, como as plantas químicas (Biegler, 2018).

Para plantas industriais de grandes dimensões, é muito comum que o problema seja decomposto em problemas menores mais simples de serem resolvidos, ainda que não-lineares (Cardoso & Kwong, 2020). A decomposição do problema pode ser determinada de maneira empírica e natural, de acordo com a própria estrutura do problema, além da separabilidade das restrições e da função objetivo (Inalhan et al., 2002). Nas plantas químicas industriais, é muito comum que a própria estrutura do processo. Porém, embora essa seja uma maneira eficaz de lidar com este tipo de problema, ainda é necessário buscar algoritmos e softwares capazes de lidar com problemas não-lineares de forma eficiente.

Ferramentas computacionais que possibilitem a resolução deste tipo de problema de modo eficaz e com tempo reduzido

resultam, em geral, em grande custo além de apresentar pouca flexibilidade ao usuário na sua aplicação. Além disso, muitos problemas de otimização precisam ser resolvidos em tempo real, o que resulta em um maior esforço computacional e potencializa a necessidade de ferramentas eficientes e precisar para lidar com este tipo de sistema (Liu & Qin, 2019).

Neste contexto, a aplicação de ferramentas como o software livre Scilab, apresenta-se uma alternativa de baixo custo para o desenvolvimento de pesquisas com enfoque na otimização não-linear de processos químicos com restrições, uma vez que este é um software gratuito e com a possibilidade de acesso ao seu código fonte, possibilitando adequação dos algoritmos disponíveis aos sistemas de interesse e até mesmo o desenvolvimento de novos algoritmos para a resolução deste tipo de problema. Sendo assim, neste artigo avalia-se a resolução de problemas de otimização de sistemas não-lineares com restrições de processos industriais, utilizando como ferramenta computacional para resolução destes o *software* livre Scilab.

2. Metodologia

Este trabalho tem natureza teórico-computacional quantitativa e foi desenvolvido de modo a possibilitar a compreensão de um problema específico, contribuindo para o desenvolvimento de novas pesquisas, através da metodologia de simulação na qual considerou-se a formulação do problema, definição do modelo, construção e verificação do programa computacional, realização das execuções de simulação e análise dos resultados como passos fundamentais para o desenvolvimento da pesquisa (Gavira, 2003; Pereira et al., 2018). A análise e validação dos resultados foram feitas a partir de comparação entre abordagens tradicionais de otimização e os métodos escolhidos e a resolução do mesmo foi feita utilizando o *software* livre Scilab, em sua versão atual (6.0.1).

2.1 Otimização não-linear com restrições para processos químicos

Um problema de otimização pode ser classificado quanto a sua linearidade, sendo que em problemas de otimização (ou programação) linear, busca-se a minimização ou maximização de uma função linear, definida sobre um domínio descrito por equações e/ou inequações lineares (Souza, 2000). Já um problema de otimização (ou programação) não linear envolve uma função que deve ser maximizada ou minimizada sujeito a um conjunto de restrições, em que pelo menos uma destas deve ter a característica de ser não linear, e em geral o ponto ótimo se encontra no contorno ou no interior do domínio da região de soluções viáveis (Cavasotti & Biehl, 2018).

A otimização não-linear (NLP) com restrições é um tipo de problema com em processos químicos industriais de grandes dimensões, podendo ser classificados como problemas de otimização não-linear inteiro mista (MINLP) quando envolvem, além de variáveis reais, variáveis discretas e a sua solução tem como objetivo, quando aplicada a este tipo de processo, orientar decisões de projeto, operação e controle baseados na resolução do problema de otimização (Biegler, 2018). A **Tabela 1** apresenta os tipos de problemas comumente resolvidos para diferentes tipos de equipamentos e etapas para a indústria química.

O problema de otimização não-linear é descrito de maneira genérica na Equação 1.

$$\min_{\mathbf{X}} f(\mathbf{X}) \tag{1}$$

sujeito a

$$h_i(\mathbf{X}) = 0, \text{ para } i = 1, 2 \dots n$$

$$g_j(\mathbf{X}) \leq 0, \text{ para } j = 1, 2, \dots, m.$$

sendo g , h e g são a função objetivo e as restrições de igualdade e desigualdade, sendo estas equações não-lineares das variáveis $\mathbf{X} \in \mathfrak{R}^{n \times}$. O objetivo da otimização global é definir \mathbf{X} de modo manter o sistema nos pontos ótimos definidos.

Tabela 1 - Tipos de problemas de otimização comuns em processos químicos industriais.

Programação Matemática em processos químicos					
	LP	MILP	QP	NLP	MINLP
Síntese e projeto de processos					
Trocadores de Calor	X	X		X	X
Separadores		X		X	X
Reatores	X			X	X
Fluxograma de processos				X	X
Operação de processos					
<i>Scheduling</i>	X	X			X
<i>Supply Chain</i>	X	X			X
Otimização em tempo real	X		X	X	
Controle de Processos					
Controle Preditivo baseado em modelo (MPC)	X		X		
MPC não-linear			X	X	
MPC híbrido		X			

Fonte: Biegler (2018).

Os métodos para a resolução de problemas de otimização não-linear com restrições são classificados em dois tipos: métodos diretos e métodos indiretos. Os métodos diretos são assim chamados por manipularem as restrições de maneira explícita enquanto os métodos indiretos resolvem o problema como uma sequência de problemas de otimização sem restrições (Rao, 2009). A resolução de problemas desta natureza é feita a partir de métodos específicos para lidar com as não-linearidades do problema dos quais destacam-se os métodos diretos Programação quadrática sucessiva (SQP) e Método dos Gradientes Reduzidos Generalizados (GRG) e o método indireto do Pontos Interiores (*interior-point*) para sistemas de grandes dimensões.

O método de Programação Quadrática Sequencial (SQP) é um dos métodos mais recentes e mais eficientes para a resolução de problemas NLP (Rao, 2009). O método baseia a busca pela solução do problema na resolução das condições de Karush-Kuhn-Tucker (KKT), que são condições para que a solução do problema não-linear seja ótima desde que as soluções de regularidade sejam satisfeitas, de modo a aproximar o problema para que a aplicação do método de Newton ou Quasi-Newton para problemas sem restrição, de convergência quadrática (Rao, 2009). O problema resultante é associado a uma técnica de região de confiança para prover uma região de busca para a solução ótima. O problema resultante é apresentado na Equação 2.

$$\min_{\mathbf{X}} Q(\mathbf{S}) = \nabla f^T \mathbf{S} + \frac{1}{2} \mathbf{S}^T [\nabla^2 L] \mathbf{S} \quad (2)$$

Sujeito a

$$h_i + \nabla h_i^T \mathbf{S} = 0, \quad i = 1, 2, \dots, n$$

$$g_j + \nabla g_j^T \mathbf{S} \leq 0, \quad j = 1, 2, \dots, m$$

A função Lagrangeana é definida na Equação 3.

$$L(\mathbf{X}, \boldsymbol{\lambda}) = f(\mathbf{X}) + \sum_{j=1}^m \lambda_j g_j(\mathbf{X}) + \sum_{i=1}^n \lambda_{i+m} h_i(\mathbf{X}) = 0 \quad (3)$$

O método possui como vantagens o menor número de cálculos de funções e gradientes, não necessitando satisfazer as restrições de igualdade ao longo do processo, sendo a solução do problema obtida com um número menor de iterações que problemas sem restrições já que estas fornecem informações para a direção de busca e passo apropriados para a resolução do problema. Em contrapartida, frequentemente viola as restrições não-lineares até a convergência (Edgar et al., 2001; Rao, 2009).

O método GRG tem como conceito básico a resolução de subproblemas a partir da eliminação de variáveis, através da adição de variáveis folga nas restrições de desigualdade do problema, podendo ser classificado como um aperfeiçoamento do método de Wolfe para problemas com restrições lineares, através da associação com estratégias do método de linearização (Santos, 1998). O problema de otimização resultante é apresentado na Equação 4.

$$\min_{\mathbf{X}} f(\mathbf{X}) \quad (5)$$

sujeito a

$$h_i(\mathbf{X}) = 0, \text{ para } i = 1, 2 \dots n$$

$$g_j(\mathbf{X}) + \mathbf{X}_{n+j} = 0, \text{ para } j = 1, 2, \dots m.$$

$$\mathbf{X}_{n+j} \geq 0, \text{ para } j = 1, 2, \dots, m$$

sendo $\mathbf{X} = \begin{bmatrix} \mathbf{Y} \\ \mathbf{Z} \end{bmatrix}$, $\mathbf{Y} = [y_1, y_2, \dots, y_n]^T$ e $\mathbf{Z} = [z_1, z_2, \dots, z_n]^T$. A Equação 5 apresenta a diferenciação da função objetivo, relacionando-a com as definições de \mathbf{Y} e \mathbf{Z} .

$$df(\mathbf{X}) = \sum_{i=1}^{n-p} \frac{\partial f}{\partial y_i} dy_i + \sum_{i=1}^{m+p} \frac{\partial f}{\partial z_i} dz_i = \nabla_{\mathbf{Y}}^T f d\mathbf{Y} + \nabla_{\mathbf{Z}}^T f d\mathbf{Z} \quad (5)$$

Através da diferenciação das restrições de desigualdade no ponto ótimo (Equação 6), no qual $d\mathbf{g} = \mathbf{0}$, obtém a definição de $d\mathbf{Z}$ (Equação 7).

$$dg_i(\mathbf{X}) = \sum_{j=1}^{n-p} \frac{\partial g_i}{\partial y_j} dy_j + \sum_{j=1}^{m+p} \frac{\partial g_i}{\partial z_j} dz_j \quad (6)$$

$$d\mathbf{g} = C d\mathbf{Y} + D d\mathbf{Z} \rightarrow d\mathbf{Z} = -D^{-1} C d\mathbf{Y} \quad (7)$$

Substituindo a Equação 7 na Equação 5, obtém-se a definição dos coeficientes generalizados reduzidos, apresentada na Equação 8.

$$\begin{aligned} df &= \nabla_Y^T f d\mathbf{Y} - \nabla_Z^T f (D^{-1} C d\mathbf{Y}) \\ \frac{df}{d\mathbf{Y}} &= G_R \\ G_R &= \nabla_Y^T f - \nabla_Z^T f (D^{-1} C) \end{aligned} \quad (8)$$

O método gera uma região de busca de maneira similar ao gradiente da função objetivo fazendo com que as direções não violem os limites das variáveis. É um dos métodos mais robustos e extremamente versátil pela facilidade de implementação, principalmente para problemas com grande número de variáveis porém, necessita que as restrições de igualdade sejam satisfeitas a cada iteração (Edgar et al., 2001; Rao, 2009).

O método dos Pontos Interiores (*Interior-Point*) foi inicialmente proposto para a resolução de problemas de programação linear de grandes dimensões por Karmarkar (Guardia, 2005), tendo sido posteriormente aplicados com sucesso na resolução de problemas de programação não-linear. Como um método indireto, este propõe a resolução de um problema de otimização com restrições através da resolução sequencial de problemas não-restritos através da inclusão de parâmetros de penalidade (r_k) de modo que o problema de otimização descrito na Equação 1 torna-se o problema descrito na Equação 9 (Rao, 2009).

$$\min_{\mathbf{X}} \Phi_k = \Phi(\mathbf{X}, r_k) = f(\mathbf{X}) + \sum_{i=1}^n h_i(\mathbf{X}) + r_k \sum_{j=1}^m G_j[g_j(\mathbf{X})] \quad (9)$$

na qual G_j é uma função da restrição g_j definida comumente, para o método dos pontos interiores como apresentado nas Equações 10 e 11.

$$G_j = -\frac{1}{g_j(\mathbf{X})} \quad (10)$$

$$G_j = \log[-g_j(\mathbf{X})] \quad (11)$$

Neste método, a barreira leva a uma região que consiste em pontos estritamente positivos de modo a determinar o vetor solução do problema. À medida que r_k decresce, a solução se aproxima da solução do problema não-linear (Equação 1). O método é considerado um caso particular do método de penalidades, trabalhando no interior da região factível, e possui diversas variações, tanto na forma primal quanto na forma dual, mantendo sempre a ideia da projeção do gradiente no espaço nulo. Possui como vantagem a rápida convergência além da robustez e eficiência pra resolução de problemas de programação não-linear (Delgado, 2021).

2.2 O software livre Scilab e a função *fmincon*

A otimização de processos químicos tem sido um campo de estudo fascinante por várias décadas. Até cerca de 1980, virtualmente, todos os problemas da engenharia química eram otimizados usando funções de objetivo único. A razão para resolver tais problemas de otimização relativamente simples foram, possivelmente, as limitações impostas pela tecnologia de sistemas computacionais (Bhaskar et al., 2000).

A criação e implantação de modelos não lineares para processos químicos são essenciais para a tomada de decisões em projetos, operações e controle, tendo ainda uma forte tendência atual que tais tarefas sejam integradas de modo a encontrar condições ótimas para o processo, o que torna os problemas de otimização ainda mais complexos (Costa et al., 2019; Patil et al., 2015; Zhou et al., 2018). Consequentemente, a qualidade dessas decisões é fortemente influenciada pela eficiência das estratégias de otimização, bem como sua interação com modelos de simulação detalhados para formar modelos de otimização integrados.

Nas últimas três décadas, foram feitos grandes avanços em algoritmos de programação não linear (PNL) e *software* para otimização de processos. Além disso, ambientes de modelagem de otimização poderosos permitem a formulação e solução de problemas de otimização em grande escala (Biegler, 2018). Porém, grande parte dos *softwares* utilizados na resolução de problemas de otimização não-lineares são pagos, de interface não amigável e possuem restrições para lidar com as exigências que este tipo de problema apresenta (Zhou et al., 2018), o que muitas vezes inviabiliza o seu uso para o desenvolvimento de pesquisas sendo necessário a busca de alternativas mais viáveis mas tão eficazes quanto os softwares pagos disponíveis.

O Scilab é um *software* livre e possui uma linguagem de programação associada a uma rica coleção de caracteres numéricos e algoritmos que cobrem diversos aspectos de problemas científicos interpretados para uma linguagem computacional. Os usuários desse *software* podem desenvolver seus próprios módulos para que eles possam resolver seus problemas particulares. Uma vez que a linguagem utilizada pelo Scilab permitir compilar e vincular dinamicamente outras linguagens, como Fortran e C, esse *software* permite também que bibliotecas externas sejam usadas como se fossem recursos internos do Scilab (Baudin, 2010).

O pacote externo *fmincon* disponível para o *software* Scilab é um solucionador de otimização multivariável não linear restrito a partir do método do ponto interior (IpOpt) que possui como autores Michael Baudin e Stéphane Motteet. Segundo os autores a função, por padrão, usa uma fórmula L-BFGS para calcular uma aproximação do hessiano do lagrangeano, gerando uma serie de iterações a partir de uma entrada de dados inicial.

O algoritmo IpOpt exibe um bom desempenho em alcançar a convergência global sob condições moderadas com altas taxas de convergência, sendo, principalmente, uma ferramenta de otimização global baseada em um método de ponto interior primal-dual (Cunha et al., 2017). O algoritmo aplica um método de barreira logarítmica para restrições de desigualdade ao problema de programação não linear (PNL), resolvendo um conjunto de problemas de otimização com restrições de igualdade para uma sequência monótona decrescente do parâmetro de barreira e rapidamente converge para a solução do PNL original. Esta sequência de problemas é resolvida com o método de Newton esparsa aplicado às condições KKT (Biegler & Zavala, 2009).

Os métodos Quasi-Newton, exigem que apenas o gradiente da função objetivo seja fornecido a cada iteração. A partir da medição das mudanças nos gradientes, eles constroem um modelo da função objetivo que é bom o suficiente para produzir convergência superlinear. O algoritmo Quase-Newton mais popular é o método BFGS, nomeado em homenagem a seus descobridores Broyden, Fletcher, Goldfarb e Shanno. Uma vantagem desse método é que ele possui propriedades de autocorreção muito eficazes (Mikosch, Resnick & Robinson, 2009).

3. Resultados e Discussão

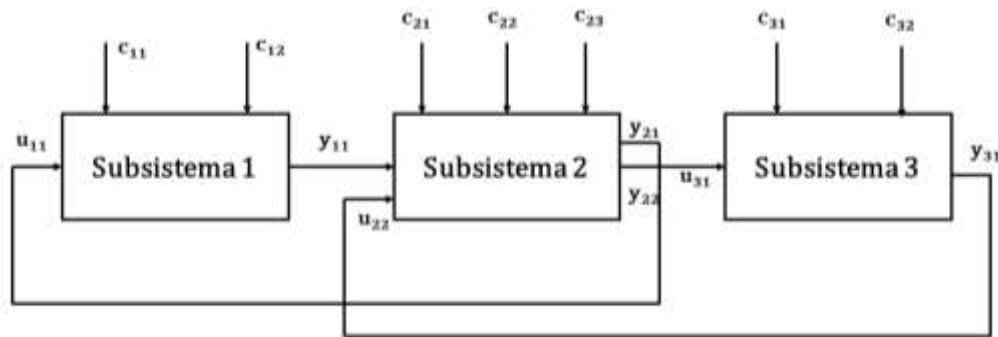
O algoritmo de otimização foi aplicado na resolução de dois problemas: um processo não-linear proposto por Findeisen et al. (1980) formado por três subsistemas interconectados e a planta de Willians-Otto (Cardoso & Kwong, 2020; Willians & Otto, 1960).

Os problemas de otimização foram implementados no *software* Scilab versão 6.1.0, utilizando o pacote de otimização fmincon versão 1.0.3, utilizando o algoritmo para otimização não-linear com o método *IpOPT*. As simulações foram realizadas em notebook com processador Intel(R) Core (TM) i5-4210U com 8GB de memória RAM e frequência de 1.70 GHz, com sistema operacional Windows 10. Além disso, os mesmos problemas foram resolvidos no *software* Matlab (versão 7-R2015a) a fim estabelecer um comparativo entre os resultados obtidos e o desempenho de cada dos softwares selecionados.

2.3 Processo não-linear composto por três subsistemas interconectados

O processo de estudo é apresentado no trabalho de Findeisen et al. (1980), composto de três subsistemas não-lineares interconectados, conforme apresentado na **Figura 1**.

Figura 1 - Três subsistemas interconectados.



Fonte: Adaptado de Kwong (1992).

O problema de otimização consiste em minimizar o índice de performance do sistema global $\Phi(c, u)$, conforme definido na Equação 12.

$$\min_{c,u} \Phi \quad (12)$$

sujeito às restrições apresentadas nas Equações 13 a 22.

$$y_{11} = c_{11} - c_{12} + 2u_{11} \quad (13)$$

$$y_{21} = c_{21} - c_{22} + u_{21} - 3u_{22} \quad (14)$$

$$y_{22} = 2c_{22} - c_{23} + u_{21} + u_{22} \quad (15)$$

$$y_{31} = c_{31} + 2,5c_{32} - 4u_{31} \quad (16)$$

$$c_{11}^2 - c_{12}^2 \leq 1 \quad (17)$$

$$0,5c_{21} + c_{22} + 2c_{23} \leq 1 \quad (18)$$

$$4u_{21}^2 + 2c_{21}u_{21} + 0,4u_{21} + c_{21}c_{23} + 0,5c_{23}^2 + u_{21} \leq 4 \quad (19)$$

$$c_{31} + u_{31} + 0,5 \geq 0 \quad (20)$$

$$0 \leq u_{11} \leq 0,5 \quad (21)$$

$$0 \leq c_{32} \leq 1 \quad (22)$$

Os autores propõem duas definições para a função objetivo, apresentadas nas Equações 23 e 24, aqui definidas como Problema 1 e 2, respectivamente.

$$\Phi = \Phi_1 + \Phi_2 + \Phi_3 \quad (23)$$

$$\Phi = \Phi_1 \cdot \Phi_2 + \Phi_3 \quad (24)$$

As funções de cada subsistema são definidas nas Equações 25 a 27.

$$\Phi_1 = ((u_{11} - 1)^4) + 5(c_{11} + c_{12} - 2)^2 \quad (25)$$

$$\Phi_2 = 2(c_{21} - 2)^2 + c_{22}^2 + 3c_{23}^2 + 4u_{21}^2 + u_{22}^2 \quad (26)$$

$$\Phi_3 = (c_{31} + 1)^2 + (u_{31} - 1)^2 + 2.5c_{32}^2 \quad (27)$$

A estimativa inicial para os métodos foi definida com base no trabalho de Findeisen et al. (1980) e os parâmetros para a função *fmincon* foram mantidos no valor padrão da função sendo a violação das restrições igual a 10^{-4} com estratégia de atualização do parâmetro de barreira definido como “monotone” com valor inicial igual a 0,1. Os resultados obtidos a partir da função *fmincon* do Scilab para o problema 1 e 2 são apresentados na Tabela 2.

Os resultados obtidos para o Problema 1 através dos dois softwares são semelhantes aos apresentados por Findeisen et al. (1980) e Kwong (1992). Porém, para o Scilab, o algoritmo necessitou de 10 iterações e 11 avaliações da função objetivo para encontrar a solução do problema, obtendo violação de restrições da ordem de 10^{-16} assim como para a resolução do problema no Matlab. Porém, neste caso, o software necessitou de 13 iterações e 228 avaliações da função objetivo para alcançar o resultado apresentado.

Para o Problema 2, ambos os softwares necessitaram de 12 iterações para encontrar o ponto ótimo para o problema, porém, enquanto o software livre Scilab realizou 13 avaliações da função objetivo, o Matlab necessitou de 214. Ambos obtiveram violação de restrições de ordem semelhante ao obtido pelo Problema 1. Os resultados encontrados para este problema se assemelham aos definidos na literatura (Findeisen et al., 1980).

Tabela 2 - Otimização de processo composto por três subsistemas interconectados.

Variável	Scilab		Matlab	
	Problema 1	Problema 2	Problema 1	Problema 2
c_{11}	0,530539	0,531602	0,53540	0,531602
c_{12}	0,847660	0,846995	0,847660	0,846994
u_{11}	0,172881	0,164570	0,172881	0,164569
c_{21}	0,990947	0,997101	0,990947	0,997100
c_{22}	-0,147284	-0,152386	-0,147284	-0,152386
c_{23}	0,003079	-0,010018	0,003079	-0,010019
u_{21}	0,028641	0,013747	0,028641	0,013747
u_{22}	0,331330	0,332888	0,331331	0,332888
c_{31}	-0,505042	-0,524388	-0,505041	-0,524385
c_{32}	0,342615	0,381930	0,342616	0,381930
u_{31}	0,005042	0,024388	0,005042	0,024388
Φ	6,100754	6,733097	6,100755	6,733098

Fonte: Autores.

2.3 Planta de Williams-Otto

A planta de fictícia apresentada no trabalho desenvolvido por Willians & Otto (1960) é um dos problemas de otimização não-linear com restrições para processos industriais mais conhecidos e foi amplamente estudada por anos (Govindarajan & Karunanithi, 2005; Jin, 1996; Vinante & Valladares, 1985). A planta é composta de um reator CSTR isotermico, um trocador de calor, um decantador e uma coluna de destilação abordando aspectos de cinética, termodinâmica, otimização, projeto e controle do processo estudado. O produto de fundo da torre é dividido em duas correntes com igual composição, sendo uma delas enviada novamente ao reator como reciclo (R) e a outra utilizada como combustível em outra etapa do processo, reduzindo o custo. A representação da planta é apresentada na

A função objetivo da otimização utilizada (Equação 28) foi a definida por Jung, Miroshi e Ray (1971), cujo problema de otimização consiste em maximizar a função objetivo que indica o retorno do investimento da planta:

$$I(\%) = 100 M - SARE - U - FCPPIN \quad (28)$$

sendo as variáveis:

- Retorno bruto por hora: $M = 8400 (P_p F_p + P_D F_D - P_A F_A - P_B F_B - P_G F_G)$
- Vendas, administração e taxas de pesquisa e engenharia: $SARE = 1041,6(P_p F_p + P_D F_D)$
- Custos fixos: $FCP = 60V_{R\rho R}$
- Custos de utilidades: $U = 2,2F_R$
- Investimentos totais: $PIN = 600V_{R\rho R}$

Figura 2.

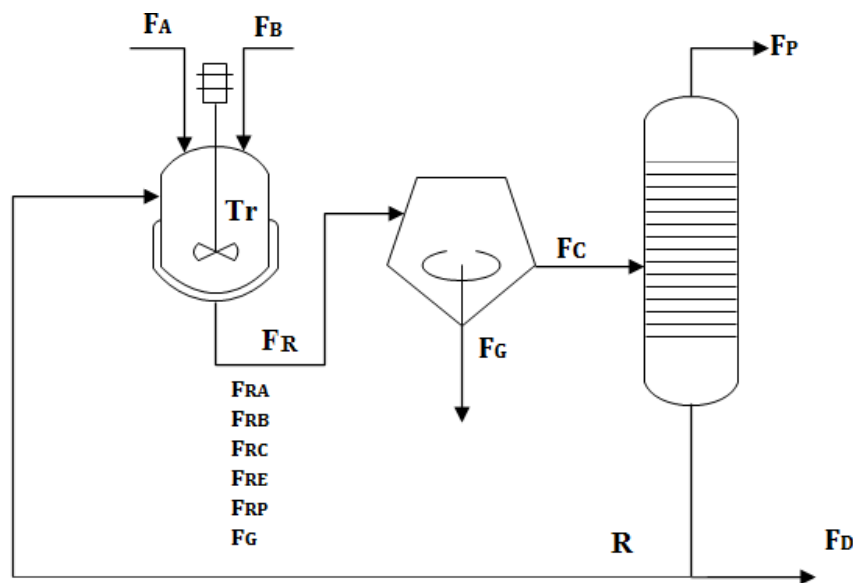
A função objetivo da otimização utilizada (Equação 28) foi a definida por Jung, Miroshi e Ray (1971), cujo problema de otimização consiste em maximizar a função objetivo que indica o retorno do investimento da planta:

$$I(\%) = 100 \frac{M - SARE - U - FCP}{PIN} \quad (28)$$

sendo as variáveis:

- Retorno bruto por hora: $M = 8400 (P_P F_P + P_D F_D - P_A F_A - P_B F_B - P_G F_G)$
- Vendas, administração e taxas de pesquisa e engenharia: $SARE = 1041,6 (P_P F_P + P_D F_D)$
- Custos fixos: $FCP = 60 V_{RDR}$
- Custos de utilidades: $U = 2,2 F_R$
- Investimentos totais: $PIN = 600 V_{RDR}$

Figura 2 - Planta de Williams-Otto.



Fonte: Adaptado de Jung et al. (1971).

Como restrições do problema de otimização têm-se a limitação da matéria-prima disponível, sendo $12400 \text{ lb/h} \leq F_A \leq 16600 \text{ lb/h}$ e $F_B \leq 56000 \text{ lb/h}$ e, a limitação de temperatura, devido a decomposição do produto, $580^\circ R \leq T_R \leq 680^\circ R$. Para o sistema, além do balanço de massa global para o decantador e a coluna, inclui-se também a definição da eficiência da separação, definida pela Equação 29.

$$F_P = F_{RP} - 0,1 F_{RE} \quad (29)$$

Para os balanços de massa e energia do sistema, este é dividido em dois subsistemas, sendo o primeiro composto por reator e decantador e o segundo pela coluna de destilação flash. As reações que ocorrem no reator, bem como suas taxas, são definidas nas

Equações 30 a 32.

$$A + B \xrightarrow{k_1} C, r_1 = k_1 \frac{F_{RA}F_{RB}}{F_R^2} \quad (30)$$

$$B + C \xrightarrow{k_2} P + E, r_2 = k_2 \frac{F_{RB}F_{RC}}{F_R^2} \quad (31)$$

$$P + C \xrightarrow{k_3} G, r_3 = k_3 \frac{F_{RP}F_{RC}}{F_R^2} \quad (32)$$

nas quais é definida por Arrhenius como $k_i = A_i e^{\frac{E}{RT_R}}$. Os balanços de massa e energia do reator, coluna e decantador seguem descritos nas equações 33 a 40 e são considerados, no estado estacionário, restrições de igualdade do sistema.

$$\frac{V_R \rho_R}{F_R} \frac{dF_{RA}}{dt} = F_A + R \left(\frac{F_{RA}}{F_R - F_G - F_P} \right) - F_{RA} - r_1 V_R \rho_R \quad (33)$$

$$\frac{V_R \rho_R}{F_R} \frac{dF_{RB}}{dt} = F_B + R \left(\frac{F_{RB}}{F_R - F_G - F_P} \right) - F_{RB} - r_1 V_R \rho_R - r_2 V_R \rho_R \quad (34)$$

$$\frac{V_R \rho_R}{F_R} \frac{dF_{RC}}{dt} = R \left(\frac{F_{RC}}{F_R - F_G - F_P} \right) - F_{RC} + 2r_1 V_R \rho_R - 2r_2 V_R \rho_R - r_3 V_R \rho_R \quad (35)$$

$$\frac{V_R \rho_R}{F_R} \frac{dF_{RE}}{dt} = R \left(\frac{F_{RE}}{F_R - F_G - F_P} \right) - F_{RE} + r_2 V_R \rho_R \quad (36)$$

$$\frac{V_R \rho_R}{F_R} \frac{dF_{RP}}{dt} = R \left(\frac{F_{RP} - F_P}{F_R - F_G - F_P} \right) - F_{RP} - r_2 V_R \rho_R - 0,5r_3 V_R \rho_R \quad (37)$$

$$\frac{V_R \rho_R}{F_R} \frac{dF_{RP}}{dt} = -F_G + 1,5r_3 V_R \rho_R \quad (38)$$

$$F_C = F_R - F_G \quad (39)$$

$$F_C = F_P + F_D + \hat{R} \quad (40)$$

Os parâmetros para o problema de otimização são apresentados em (Cardoso, 2016). Para a resolução do problema, utilizou-se como estimativa inicial para os valores ótimos apresentados no trabalho original de Jung et al. (1971). Utilizou-se para aproximação das diferenças finitas a opção “ComplexStep” que utiliza o Método de Passo Complexo de ordem 2. O problema também foi resolvido *software* Matlab (versão 7-R2015a), utilizando o mesmo algoritmo. Os resultados obtidos pelo Scilab são apresentados na Tabela 3.

Tabela 3 - Resultado da otimização da Planta de Williams-Otto.

Variável	Matlab	Scilab
F_A	13548,5249	12522,2223
F_B	31504,9455	28358,2790
F_R	157300,7994	326888,9047
F_{RA}	18186,9801	38518,7926
F_{RB}	60816,1800	121864.0808
F_{RC}	3336,7449	7315,9849

F_{RE}	60538,7030	137745,08504
F_{RP}	10816,8703	18537,5085
F_G	3605,32127	2907,4529
F_D	36685,14909	33210, 0482
T_R	655,84667	654,2169
I	72,36577	98,6785

Fonte: Autores.

Pelos resultados obtidos, é possível verificar que o ponto ótimo obtido pelo Scilab possibilita um aumento de cerca de 16% no retorno do investimento da planta, utilizando para isso 56 iterações e 62 avaliações da função enquanto o Matlab utiliza apenas 3 iterações e 24 avaliações da função objetivo. As violações das restrições são de $1,235 \cdot 10^{-10}$ e $7,5317 \cdot 10^{-9}$ para o Scilab e o Matlab, respectivamente.

Há também uma diferença significativa entre o número de iterações necessárias, o tempo de processamento para resolução do problema é semelhante: para o Scilab são necessários, aproximadamente, 5,49 s para encontrar a solução ótima, enquanto para o Matlab a solução ótima é obtida em 0,629s, representando um tempo de processamento de mais de 10 vezes menor para o software pago.

Porém, utilizando a mesma estimativa inicial, o Matlab encontra um ótimo local, conforme as informações de processamento obtidas pela própria função. Ao utilizar o ponto obtido pelo Scilab como estimativa inicial, o algoritmo consegue convergir para o ótimo global, obtendo a mesma resposta obtida pelo Scilab, o que não acontece com o Scilab ao utilizarmos ponto ótimo obtido pelo Matlab como estimativa inicial para a resolução do problema: o software livre continua encontrando o mesmo ponto ótimo apresentado na Tabela 3.

4. Conclusões

A programação não-linear é um tipo de problema de otimização que abrange diversas áreas da engenharia como sistemas de energia, sistemas de controle. Somada a não-linearidade da função objetivo, as restrições, sendo estas lineares ou não-lineares, podem tornar o problema de difícil resolução e a busca por ferramentas que consigam lidar com estas características de forma eficaz para se determinar as condições ótimas de operação, torna-se uma exigência para o desenvolvimento de novas pesquisas e tecnologias.

O software livre Scilab tem disponível a função `fmincon`, cujo algoritmo `IpOpt` é utilizado para definir pontos ótimos de operação para problemas não-lineares com restrições. Os testes realizados neste trabalho, utilizando problemas clássicos da Engenharia da programação não-linear restritos mostra a eficiência do software. Os resultados obtidos se assemelham aos resultados da literatura para o primeiro exemplo e quando comparados ao ponto ótimo obtido pelo Matlab é possível verificar que o problema foi resolvido de forma satisfatória.

Para a planta de Williams-Otto, o desempenho do Scilab foi ainda melhor, apresentando um aumento de cerca de 16% na função objetivo. É possível verificar que o ótimo encontrado foi o ótimo global independente da estimativa inicial utilizada, o que não ocorre quando o problema é resolvido através do Matlab que, apenas consegue encontrar o resultado semelhante quando se altera

a estimativa inicial para um valor mais próximo deste, encontrando um ótimo local que apresenta função objetivo menor ao utilizar-se a estimativa do trabalho dos autores Jung et al.(1971). Porém, é importante ressaltar que o esforço computacional para a resolução do problema no Scilab é maior, o que aumenta de forma considerável o tempo quando comparado ao Matlab. No caso deste problema específico, mesmo com um aumento percentual tão expressivo, considerando a escala de tempo necessária para a resolução do problema, não pode ser um fator relevante ao analisar o desempenho do software na resolução deste tipo de problema.

Sendo assim, o software livre apresentou-se como uma alternativa viável e eficaz na resolução de problemas de otimização não-linear com restrições, sendo este uma importante ferramenta no desenvolvimento de pesquisas desta natureza. É importante ressaltar que a função possui apenas uma opção de algoritmo para a resolução deste tipo de problema, o que pode torna-la limitada, mas considerando o desenvolvimento do software nos últimos tempos, tem se tornado uma alternativa promissora.

Como sugestões para trabalhos futuros, a função pode ser utilizada para a resolução de problemas de dimensões maiores e para problemas de otimização dinâmica, cuja complexidade permitirá avaliar se há restrições quanto a aplicação do software em processos de alta complexidade.

Referências

- Baudin, M. (2010). Introduction to scilab. *Consortium Scilab*, 34–56.
- Bhaskar, V., Gupta, S. K., & Ray, A. K. (2000). Applications of multiobjective optimization in chemical engineering. *Reviews in Chemical Engineering*, 16(1), 1–54.
- Biegler, L. T. (2018). New directions for nonlinear process optimization. *Current Opinion in Chemical Engineering*, 21, 32–40.
- Cardoso, A. O. (2016). *Otimização Descentralizada Coordenada aliada a estratégias de Controle Plantwide para o Controle de Processos Químicos*. Universidade Federal de São Carlos (UFSCar).
- Cardoso, A. O., & Kwong, W. H. (2020). Estrutura de controle plantwide simplificada para a planta de Williams-Otto: otimização descentralizada coordenada com IBMF. *Research, Society and Development*, 9(9), 1–21.
- Chaudhary, M. N. R. (2009). *Real Time Optimization of Chemical Processes*. Curtin University of Technology.
- Costa, E. P., Figueiredo, M., Politano, P. R., & Kwong, W. H. (2019). O impacto na programação da produção devido à integração das camdas de controle avançado e de scheduling na indústria de processos. *Brazilian Journal of Business*, 1(3), 1715–1727.
- Delgado, J. A. (2021). *Método primal-dual de pontos interiores-penalidade e o problema de fluxo de potência ótimo reativo com variáveis de controle discretas*. Universidade Estadual Paulista _Júlio de Mesquita Filho_.
- Edgar, T. F., Himmelblau, D. M., & Lasdon, L. S. (2001). *Optimization of Chemical Processes*. McGraw-Hill.
- Findeisen, W., Bailey, F. N., Brdys, M., Malinowski, K., Tatjewski, P., & Wäsniak, A. (1980). *Control and Coordination in Hierarchical Systems*. Jonh Wiley & Sons.
- Gavira, M. de O. (2003). *Simulação como ferramenta computacional de aquisição de conhecimento*. UFSCar.
- Govindarajan, L., & Karunanithi, T. (2005). Multiobjective optimization of process plant using genetic algorithm. *International Journal of Computational Intelligence and Applications*, 5(04), 425–437.
- Guardia, L. E. T. (2005). Método dos Pontos Interiores para programação não-linear em redes. *XXXVII Simpósio Brasileiro de Pesquisa Operacional*, 1930–1936.
- Inalhan, G., Stipanović, D. M., & Tomlin, C. J. (2002). Decentralized optimization, with application to multiple aircraft coordination. *Proceedings of the IEEE Conference on Decision and Control*, 1, 1147–1155. <https://doi.org/10.1109/cdc.2002.1184667>
- Jin, X. (1996). *Approaching sustainability in engineering design with multiple criteria decision analysis*. Oklahoma State University.
- Jung, B. S., Miroshi, W., & Ray, W. H. (1971). Large Scale Process Optimization Techniques Applied to Chemical Petroleum Processes. *The Canadian Journal of Chemical Engineering*, 844–852.
- Kwong, W. H. (1992). *Otimizaç{ão}de Plantas Industriais Complexas*. Escola Politécnica da Universidade de S{ão}Paulo (USP).

- Liu, N., & Qin, S. (2019). A neurodynamic approach to nonlinear optimization problems with affine equality and convex inequality constraints. *Neural Networks*, 109, 147–158. <https://doi.org/10.1016/j.neunet.2018.10.010>
- Miller, D. C., Siirola, J. D., Agarwal, D., Burgard, A. P., Lee, A., Eslick, J. C., Nicholson, B., Laird, C., Biegler, L. T., Bhattacharyya, D., Sahinidis, N. V., Ignacio E. Grossmann, Gounaris, C. E., & Gunter, D. (2018). Next Generation Multi-Scale Process Systems Engineering Framework. *Proceedings of the 13th International Symposium on Process Systems Engineering – PSE 2018*, 2209–2214.
- Patil, B. P., Maia, E., & Ricardez-Sandoval, L. A. (2015). Integration of Scheduling, Design and Control of Multiproduct Chemical Processes under Uncertainty. *AIChE Journal*, 61(8), 2456–2470.
- Pereira, A. S., Shitsuka, D. M., Parreira, F. J., & Shitsuka, R. (2018). *Metodologia da pesquisa científica [recurso eletrônico]* (1ª edição). NTE/UFSM. https://repositorio.ufsm.br/bitstream/handle/1/15824/Lic_Computacao_Metodologia-Pesquisa-Cientifica.pdf?sequence=1
- Rangaiah, G. P., & Kariwala, V. (2012). *Plantwide Control: recent developments and Applications*. John Wiley & Sons.
- Rao, S. S. (2009). *Engineering Optimization: Theory and Practice* (John Wiley and Sons (ed.) (4a ed.).
- Ryoo, H. S., & Sahinidis, N. V. (1995). Global optimization of nonconvex NLPs and MINLPs with applications in process design. *Computers and Chemical Engineering*, 19(5), 551–566. [https://doi.org/10.1016/0098-1354\(94\)00097-2](https://doi.org/10.1016/0098-1354(94)00097-2)
- Santos, T. J. G. dos. (1998). *Um novo algoritmo de penalização hiperbólica para resolução de problema de programação não-linear com restrições de igualdade*.
- Thakur, S. S., Ojasvi, Kumar, V., & Nitin, K. (2017). Continuous diisobutylene manufacturing: Conceptual process design and plantwide control. *Computers and Chemical Engineering*, 97(59–75).
- Vinante, C., & Valladares, E. (1985). Application of the method of multipliers to the optimization of chemical processes. *Computers & Chemical Engineering*, 9(1), 83–87.
- Williams, T. J., & Otto, R. E. (1960). A Generalized Chemical Processing Model for the Investigation of Computer Control. *Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics*, 79(5), 458–473.
- Zhou, M., Cai, Y., Su, H., Wozny, G., & Pan, H. (2018). A survey on applications of optimization-based integrating process design and control for chemical processes. *Chemical Engineering Communications*, 205(10), 1365–1383.
- Williams, T. J., & Otto, R. E. (1960). A generalized chemical processing model for the investigation of computer control. *Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics*, 79(5), 458–473.
- Jung, B. S., Mirosh, W., & Ray, W. H. (1971). Large scale process optimization techniques applied to chemical and petroleum processes. *The Canadian Journal of Chemical Engineering*, 49(6), 844–852.