# Códigos desenvolvidos em MATLAB®

Esta seção reporta alguns códigos MATLAB desenvolvidos para a construção dos resultados, pertinentes a Tese de Doutorado intitulada “Análise 4E de uma biorefinaria produzindo biodiesel de microalgas em sinergia com a indústria sucroalcooleira no Brasil”.

A seção 1 resolve um sistema algébrico linear, de modo a se obter os coeficientes estequiométricos das moléculas participantes na reação de biofixação. Os dados de entrada são a composição em base mássica das microalgas secas.

A seção 2 utiliza os dados dos ensaios (experimentos) simulados, conforme o planejamento de Doehlert, dentro do contexto da biorefinaria arrojada. O código é exemplificado para o caso de X3 = 0 (centro do teor de lipídeos), sendo capaz de plotar as superfícies de respostas singulares como função de X1 e X2.

A seção 3 enfatiza as construções das funções desejabilidades globais, sua plotagem para casos particulares, e a otimização dentro das restrições impostas sobre os domínios.

**1. Cálculo de coeficientes estequiométricos segundo a composição em base massa dos macronutrientes**

% Reacao de referência para resolução do sistema linear CaHbNcOd -> x C57H104O6 + y C18H34O2 + z C9H11NO2 + w C12H22O11

% Dados de referencia, baseline: 38,7% TAG; 4,3% FA; 6% Proteína; 51% Carboidrato

clear; clc;

syms x y z w

TAGs = 38.7;

FAs = 4.3;

Protein = 6;

Carbs = 51;

% xp, yp, zp, wp -> frações molares calculadas a partir da composição mássica

a=1;

xp = (TAGs/884) / (TAGs/884 + FAs/282 + Protein/165 + Carbs/342)

yp = (FAs/282)/ (TAGs/884 + FAs/282 + Protein/165 + Carbs/342)

zp = (Protein/165) /(TAGs/884 + FAs/282 + Protein/165 + Carbs/342)

wp = (Carbs/342) /(TAGs/884 + FAs/282 + Protein/165 + Carbs/342)

eqn1 = xp \* (x+y+z+w) == x;

eqn2 = yp\* (x+y+z+w)== y;

eqn3 = zp \*(x+y+z+w)== z;

a=1;

eqn5 = 57\*x +18\*y + 9\*z + 12\*w == a;

sol = solve([eqn1, eqn2, eqn3, eqn5], [x, y, z, w]);

xSol = vpa(sol.x)

ySol = vpa(sol.y)

zSol = vpa(sol.z)

wSol = vpa(sol.w)

b = 104\*xSol + 34\*ySol + 11\*zSol + 22\*wSol

c = zSol

d = 6\*xSol + 2\*ySol + 2\*zSol + 11\*wSol

syms global\_H2O global\_O2

eqnalfa = global\_H2O\*2+c\*3 == b;

eqnbeta = 2 + global\_H2O == d + global\_O2\*2;

sol\_global = solve([eqnalfa,eqnbeta],[global\_H2O,global\_O2]);

global\_H2OSol = sol\_global.global\_H2O

global\_O2Sol = sol\_global.global\_O2

**2. Construção das figuras para X3 = 0 (centro do experimento para o teor de lipídeos)**

%Este matlab:

% (1) Recebe os valores de X\_reais e X\_reduzidos e Y (respostas)

% (2) Plota as respostas em termos de das variaveis reais, considerando a

% principio o ponto X3=0 (x3 =43%)para o óleo, que seria o centro do

% experimento para x3

clear;clc;

X\_reduzidos = [1 0 0 0 0 0 0 0 0 0

1 -1 1 0 0 0 0 0 0 0

1 -0.5 0.25 0.866 0.749956 0 0 -0.433 0 0

1 0.5 0.25 0.866 0.749956 0 0 0.433 0 0

1 1 1 0 0 0 0 0 0 0

1 0.5 0.25 -0.866 0.749956 0 0 -0.433 0 0

1 -0.5 0.25 -0.866 0.749956 0 0 0.433 0 0

1 0.5 0.25 0.288 0.082944 0.816 0.665856 0.144 0.408 0.235008

1 0 0 -0.577 0.332929 0.816 0.665856 0 0 -0.470832

1 -0.5 0.25 0.288 0.082944 0.816 0.665856 -0.144 -0.408 0.235008

1 0.5 0.25 -0.288 0.082944 -0.816 0.665856 -0.144 -0.408 0.235008

1 0 0 0.577 0.332929 -0.816 0.665856 0 0 -0.470832

1 -0.5 0.25 -0.288 0.082944 -0.816 0.665856 0.144 0.408 0.235008

1 0 0 0 0 0 0 0 0 0

1 0 0 0 0 0 0 0 0 0

];

X\_reais=[1 50 2500 0.58 0.3364 43 1849 29 2150 24.94

1 0 0 0.58 0.3364 43 1849 0 0 24.94

1 25 625 0.725 0.525625 43 1849 18.125 1075 31.175

1 75 5625 0.725 0.525625 43 1849 54.375 3225 31.175

1 100 10000 0.58 0.3364 43 1849 58 4300 24.94

1 75 5625 0.435 0.189225 43 1849 32.625 3225 18.705

1 25 625 0.435 0.189225 43 1849 10.875 1075 18.705

1 75 5625 0.628 0.394384 63 3969 47.1 4725 39.564

1 50 2500 0.483 0.233289 63 3969 24.15 3150 30.429

1 25 625 0.628 0.394384 63 3969 15.7 1575 39.564

1 75 5625 0.532 0.283024 23 529 39.9 1725 12.236

1 50 2500 0.677 0.458329 23 529 33.85 1150 15.571

1 25 625 0.532 0.283024 23 529 13.3 575 12.236

1 50 2500 0.58 0.3364 43 1849 29 2150 24.94

1 50 2500 0.58 0.3364 43 1849 29 2150 24.94

];

%matriz de respostas, onde cada linha é um experimento (ver X\_reduzidos por

%exemplo) e cada coluna é uma resposta, na sequencia:

% NER, FER, NEB, FEB, Ex, IC, AbtCO2, VPL, TIR

Y = [0.960044451 10.76423542 -0.175336288 3.821563712 0.311393376 15.95842691 276.6383395 82.01 0.124458313

0.996419668 10.80101281 -0.015764545 3.981135455 0.317547419 14.82775917 284.0057354 134.03 0.140855452

1.091735047 10.42682159 0.409536075 4.406436075 0.324819126 15.3826478 319.7094534 167.07 0.141209428

1.051564773 10.44502449 0.229171327 4.226071327 0.319623233 16.37920731 311.2369993 123.22 0.130483902

0.945057824 10.97555426 -0.240288604 3.756611396 0.285434974 16.81430176 273.2554912 57.20 0.116872822

0.829006021 10.90981921 -0.739649705 3.257250295 0.292916677 17.04675676 234.6183804 -0.38 0.099859531

0.87 11.08274004 -0.58 3.420204013 0.301866318 15.5143162 242.213479 45.07 0.117026854

1.065725172 10.70919253 0.291728275 4.288628275 0.346027879 14.6707564 324.172234 234.89 0.166777186

0.953581591 10.8994396 -0.20331782 3.79358218 0.328521797 14.67740221 280.1133149 155.20 0.154075455

1.117208543 10.78830057 0.522588769 4.519488769 0.356549868 13.49011121 337.7942762 296.05 0.184625216

0.740851295 10.64668391 -1.113257684 2.883642316 0.254324242 20.0043814 196.8967334 -166.89 0.037808685

0.765166645 10.84052144 -1.00988725 2.98701275 0.260449477 18.98241325 202.2111505 -134.11 0.050611812

0.780227143 10.94899597 -0.945808598 3.051091402 0.263326257 18.18981333 204.3211423 -113.28 0.058061329

0.96075522 10.74926575 -0.172253229 3.824646771 0.311400127 15.97005495 276.9560435 82.31 0.124498313

0.959562466 10.769517 -0.177434135 3.819465865 0.311333579 15.95840628 276.4679271 81.69 0.124382423

];

b\_reduzidos =load('b\_matrix.txt'); %carrega os b\_reduzidos encontrados do Statistica

b\_reais = (X\_reais'\*X\_reais)\X\_reais'\*X\_reduzidos\*b\_reduzidos; %recuperacao dos b\_reais por operacoes matriciais, dado que Y=X\_reduzidos\*b\_reduzidos=X\_reais\*b\_reais, e uso da propriedade de matrizes simétricas serem inversíveis

%Plotagens

W = 43;

[X,Y] = meshgrid(0:2:100, 0.41256351:(0.74743649-0.41256351)/50:0.74743649);

%NER

for parametros = 1:10

b(parametros) = b\_reais(parametros,1);

end

Z\_NER = b(1) + b(2)\*X + b(3)\*X.^2 + b(4)\*Y + b(5)\*Y.^2 + b(6)\*W + b(7)\*W^2 + b(8)\*X.\*Y + b(9)\*X\*W+b(10)\*Y\*W;

%FER

for parametros = 1:10

b(parametros) = b\_reais(parametros,2);

end

Z\_FER = b(1) + b(2)\*X + b(3)\*X.^2 + b(4)\*Y + b(5)\*Y.^2 + b(6)\*W + b(7)\*W^2 + b(8)\*X.\*Y + b(9)\*X\*W+b(10)\*Y\*W;

%NEB

for parametros = 1:10

b(parametros) = b\_reais(parametros,3);

end

Z\_NEB = b(1) + b(2)\*X + b(3)\*X.^2 + b(4)\*Y + b(5)\*Y.^2 + b(6)\*W + b(7)\*W^2 + b(8)\*X.\*Y + b(9)\*X\*W+b(10)\*Y\*W;

%FEB

for parametros = 1:10

b(parametros) = b\_reais(parametros,4);

end

Z\_FEB = b(1) + b(2)\*X + b(3)\*X.^2 + b(4)\*Y + b(5)\*Y.^2 + b(6)\*W + b(7)\*W^2 + b(8)\*X.\*Y + b(9)\*X\*W+b(10)\*Y\*W;

%Ex

for parametros = 1:10

b(parametros) = b\_reais(parametros,5);

end

Z\_Ex = b(1) + b(2)\*X + b(3)\*X.^2 + b(4)\*Y + b(5)\*Y.^2 + b(6)\*W + b(7)\*W^2 + b(8)\*X.\*Y + b(9)\*X\*W+b(10)\*Y\*W;

%IC

for parametros = 1:10

b(parametros) = b\_reais(parametros,6);

end

Z\_IC = b(1) + b(2)\*X + b(3)\*X.^2 + b(4)\*Y + b(5)\*Y.^2 + b(6)\*W + b(7)\*W^2 + b(8)\*X.\*Y + b(9)\*X\*W+b(10)\*Y\*W;

%AbtCO2

for parametros = 1:10

b(parametros) = b\_reais(parametros,7);

end

Z\_AbtCO2 = b(1) + b(2)\*X + b(3)\*X.^2 + b(4)\*Y + b(5)\*Y.^2 + b(6)\*W + b(7)\*W^2 + b(8)\*X.\*Y + b(9)\*X\*W+b(10)\*Y\*W;

%VPL

for parametros = 1:10

b(parametros) = b\_reais(parametros,8);

end

Z\_VPL = b(1) + b(2)\*X + b(3)\*X.^2 + b(4)\*Y + b(5)\*Y.^2 + b(6)\*W + b(7)\*W^2 + b(8)\*X.\*Y + b(9)\*X\*W+b(10)\*Y\*W;

%TIR

for parametros = 1:10

b(parametros) = b\_reais(parametros,9);

end

Z\_TIR = b(1) + b(2)\*X + b(3)\*X.^2 + b(4)\*Y + b(5)\*Y.^2 + b(6)\*W + b(7)\*W^2 + b(8)\*X.\*Y + b(9)\*X\*W+b(10)\*Y\*W;

%Plotagem sumária dos casos para X3 = 43% oleo, conforme programado acima

i=1;

subplot(3,3,i)

surf(Y,X,Z\_NER, 'edgecolor', 'none')

title('NER')

xlabel('x\_2 (-)')

ylabel('x\_1 (%)')

i=i+1;

subplot(3,3,i)

surf(Y,X,Z\_FER, 'edgecolor', 'none')

title('FER')

xlabel('x\_2 (-)')

ylabel('x\_1 (%)')

i=i+1;

subplot(3,3,i)

surf(Y,X,Z\_NEB, 'edgecolor', 'none')

title('NEB (GJ/t\_c\_a\_n\_a)')

xlabel('x\_2 (-)')

ylabel('x\_1 (%)')

i=i+1;

subplot(3,3,i)

surf(Y,X,Z\_FEB, 'edgecolor', 'none')

title('FEB (GJ/t\_c\_a\_n\_a)')

xlabel('x\_2 (-)')

ylabel('x\_1 (%)')

i=i+1;

subplot(3,3,i)

surf(Y,X,Z\_Ex, 'edgecolor', 'none')

title('Eficiência exergética (\eta)')

xlabel('x\_2 (-)')

ylabel('x\_1 (%)')

i=i+1;

subplot(3,3,i)

surf(Y,X,Z\_IC, 'edgecolor', 'none')

title('IC (gCO\_2/MJ)')

xlabel('x\_2 (-)')

ylabel('x\_1 (%)')

i=i+1;

subplot(3,3,i)

surf(Y,X,Z\_AbtCO2, 'edgecolor', 'none')

title('AbtCO2 (kgCO\_2/t\_c\_a\_n\_a)')

xlabel('x\_2 (-)')

ylabel('x\_1 (%)')

i=i+1;

subplot(3,3,i)

surf(Y,X,Z\_VPL, 'edgecolor', 'none')

title('VPL (milhões USD)')

xlabel('x\_2 (-)')

ylabel('x\_1 (%)')

i=i+1;

subplot(3,3,i)

surf(Y,X,Z\_TIR, 'edgecolor', 'none')

title('TIR (%a.a.)')

xlabel('x\_2 (-)')

ylabel('x\_1 (%)')

i=i+1;

**3. Montagem das funções desejabilidades globais e otimização**

%implementando uma restricao de dominio num circulo

function [c, ceq] = circlecon(x)

c = (x(1)^2 + x(2)^2 + x(3)^2) - 1;

ceq=[];

clear;clc;

%Este matlab:

% (1) pega os coeficientes reduzidos, gerados no Statistica, de um txt

% (2) otimiza as respostas individualmente

% (3) gera as funções de desejabilidade %(lineares) considerando plateous

% maximos e minimos

% (4) constroi a desejabilidade global de duas maneiras: pesos iguais para

% as categorias e arbitragem de especialistas

%PC de casa

% (5) otimiza essas desejabilidades globais dentro das restrições com a

% fmincon

% (6) plota as desejabildiades globais em termos de variaveis reduzidas

b= load('b\_matrix.txt'); %carrega o txt com o coeficientes dos modelos de RSM

Extremos = load('Y\_extremos.txt'); %carrega com os valore extermos observados para os indicadores, maximos linha 1; minimos linha 2

Y\_maximos = Extremos(1,:)'\*1.1; %coloca margem de 10% sobre os valores maximos

Y\_minimos = Extremos(2,:)';

% cada coluna se refere a uma resposta, ex: TIR, e cada linha se refere a

% sequencia de b obtida do Statistica, ou seja:

% b0, b1, b11, b2, b22, b3, b33, b12, b13, b23 = 10 coeficientes no total

% a sequencia de respotas eh

% NER, FER, NEB, FEB, Ex, IC, AbatCO2, VPL,TIR = 9 respostas no total,

% baseadas nas variaveis reduzidas

%Capta os coeficientes de cada resposta

b\_NER = b(:,1);

b\_FER =b(:,2);

b\_NEB =b(:,3);

b\_FEB =b(:,4);

b\_Ex =b(:,5);

b\_IC =b(:,6);

b\_AbtCO2 =b(:,7);

b\_VPL =b(:,8);

b\_TIR =b(:,9);

%Parâmetros solicitar pela otimizacao da funcao fmincon

x0 = [1/2,-1/2,0]; %Ponto de partida, caso 10: [0.5 0.288 0.816]

lb = [-1,-1,-1]; %limites inferiores das coordenadas

ub = [1,1,1]; % limites superiores das coordenadas

A = []; %parâmetro da fmincon

b = []; %parâmetro da fmincon

Aeq =[]; %parâmetro da fmincon

beq=[]; %parâmetro da fmincon

nonlcon=@circlecon3d; %possibilidade de restrição, i.e., esfera de raio 1

% Respostas no formato de função do MATLAB, útil para uso com a fmincon

fun\_NER = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_NER;

fun\_FER = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_FER;

fun\_NEB = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_NEB;

fun\_FEB = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_FEB;

fun\_Ex = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_Ex;

fun\_IC = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_IC;

fun\_AbtCO2 = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_AbtCO2;

fun\_VPL = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_VPL;

fun\_TIR = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_TIR;

%Funcoes para maximizar e minimizar as respostas na fmincon

Min\_fun\_NER = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_NER;

Min\_fun\_FER = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_FER;

Min\_fun\_NEB = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_NEB;

Min\_fun\_FEB = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_FEB;

Min\_fun\_Ex = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_Ex;

Min\_fun\_IC = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_IC;

Min\_fun\_AbtCO2 = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_AbtCO2;

Min\_fun\_VPL = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_VPL;

Min\_fun\_TIR = @(x) [1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_TIR;

Max\_fun\_NER = @(x) -1\*[1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_NER;

Max\_fun\_FER = @(x) -1\*[1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_FER;

Max\_fun\_NEB = @(x) -1\*[1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_NEB;

Max\_fun\_FEB = @(x) -1\*[1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_FEB;

Max\_fun\_Ex = @(x) -1\*[1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_Ex;

Max\_fun\_IC = @(x) -1\*[1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_IC;

Max\_fun\_AbtCO2 = @(x) -1\*[1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_AbtCO2;

Max\_fun\_VPL = @(x) -1\*[1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_VPL;

Max\_fun\_TIR = @(x) -1\*[1 x(1) x(1)^2 x(2) x(2)^2 x(3) x(3)^2 x(1)\*x(2) x(1)\*x(3) x(2)\*x(3)] \*b\_TIR;

%Definição do objetivo de otimização individual de cada resposta

flag\_minimizar\_NER = -1;

flag\_minimizar\_FER = -1;

flag\_minimizar\_NEB = -1;

flag\_minimizar\_FEB = -1;

flag\_minimizar\_Ex = -1;

flag\_minimizar\_IC = 1;

flag\_minimizar\_AbtCO2 = -1;

flag\_minimizar\_VPL = -1;

flag\_minimizar\_TIR = -1;

%Obtenção dos valores maximos e minimos das respostas

x\_min\_NER = fmincon(Min\_fun\_NER,x0,A,b,Aeq,beq,lb,ub); NER\_min = fun\_NER(x\_min\_NER);

x\_max\_NER = fmincon(Max\_fun\_NER,x0,A,b,Aeq,beq,lb,ub); NER\_max = fun\_NER(x\_max\_NER);

x\_min\_FER = fmincon(Min\_fun\_FER,x0,A,b,Aeq,beq,lb,ub); FER\_min = fun\_FER(x\_min\_FER);

x\_max\_FER = fmincon(Max\_fun\_FER,x0,A,b,Aeq,beq,lb,ub); FER\_max = fun\_FER(x\_max\_FER);

x\_min\_NEB = fmincon(Min\_fun\_NEB,x0,A,b,Aeq,beq,lb,ub); NEB\_min = fun\_NEB(x\_min\_NEB);

x\_max\_NEB = fmincon(Max\_fun\_NEB,x0,A,b,Aeq,beq,lb,ub); NEB\_max = fun\_NEB(x\_max\_NEB);

x\_min\_FEB = fmincon(Min\_fun\_FEB,x0,A,b,Aeq,beq,lb,ub); FEB\_min = fun\_FEB(x\_min\_FEB);

x\_max\_FEB = fmincon(Max\_fun\_FEB,x0,A,b,Aeq,beq,lb,ub); FEB\_max = fun\_FEB(x\_max\_FEB);

x\_min\_Ex = fmincon(Min\_fun\_Ex,x0,A,b,Aeq,beq,lb,ub); Ex\_min = fun\_Ex(x\_min\_Ex);

x\_max\_Ex = fmincon(Max\_fun\_Ex,x0,A,b,Aeq,beq,lb,ub); Ex\_max = fun\_Ex(x\_max\_Ex);

x\_min\_IC = fmincon(Min\_fun\_IC,x0,A,b,Aeq,beq,lb,ub); IC\_min = fun\_IC(x\_min\_IC);

x\_max\_IC = fmincon(Max\_fun\_IC,x0,A,b,Aeq,beq,lb,ub); IC\_max = fun\_IC(x\_max\_IC);

x\_min\_AbtCO2 = fmincon(Min\_fun\_AbtCO2,x0,A,b,Aeq,beq,lb,ub); AbtCO2\_min = fun\_AbtCO2(x\_min\_AbtCO2);

x\_max\_AbtCO2 = fmincon(Max\_fun\_AbtCO2,x0,A,b,Aeq,beq,lb,ub); AbtCO2\_max = fun\_AbtCO2(x\_max\_AbtCO2);

x\_min\_VPL = fmincon(Min\_fun\_VPL,x0,A,b,Aeq,beq,lb,ub); VPL\_min = fun\_VPL(x\_min\_VPL);

x\_max\_VPL = fmincon(Max\_fun\_VPL,x0,A,b,Aeq,beq,lb,ub); VPL\_max = fun\_VPL(x\_max\_VPL);

x\_min\_TIR = fmincon(Min\_fun\_TIR,x0,A,b,Aeq,beq,lb,ub); TIR\_min = fun\_TIR(x\_min\_TIR);

x\_max\_TIR = fmincon(Max\_fun\_TIR,x0,A,b,Aeq,beq,lb,ub); TIR\_max = fun\_TIR(x\_max\_TIR);

%Construção das funções de desejabilidade "retas"

%se for maximizar a respota (d=1 para maximo e d=0 para minimo) vale a equação

%d=(y-min)/(max-min); e se for minimizar, tem-se d=(y-max)/(min-max)

NER\_max\_casos=Y\_maximos(1); NER\_min\_casos = Y\_minimos(1);

FER\_max\_casos=Y\_maximos(2); FER\_min\_casos = Y\_minimos(2);

NEB\_max\_casos=Y\_maximos(3); NEB\_min\_casos = Y\_minimos(3);

FEB\_max\_casos=Y\_maximos(4); FEB\_min\_casos = Y\_minimos(4);

Ex\_max\_casos=Y\_maximos(5); Ex\_min\_casos = Y\_minimos(5);

IC\_max\_casos=Y\_maximos(6); IC\_min\_casos = Y\_minimos(6);

AbtCO2\_max\_casos=Y\_maximos(7); AbtCO2\_min\_casos = Y\_minimos(7);

VPL\_max\_casos=Y\_maximos(8); VPL\_min\_casos = Y\_minimos(8);

TIR\_max\_casos=Y\_maximos(9); TIR\_min\_casos = Y\_minimos(9);

%Funcao de referencia para gerar plateaus func\_PM = @(x) (x<min\_PM).\*1+(x>=min\_PM & x <= max\_PM).\*((max\_PM-x)./(max\_PM-min\_PM)) + (x>max\_PM).\*0;

desej\_NER=@(x) (fun\_NER(x)<NER\_min\_casos).\*0+ (fun\_NER(x)>=NER\_min\_casos & fun\_NER(x)<=NER\_max\_casos).\*(fun\_NER(x) - NER\_min\_casos)/(NER\_max\_casos-NER\_min\_casos)+(fun\_NER(x)>NER\_max\_casos).\*1; %maximizar

desej\_FER=@(x) 1;%(fun\_FER(x)<FER\_min\_casos).\*0+ (fun\_FER(x)>=FER\_min\_casos & fun\_FER(x)<=FER\_max\_casos).\*(fun\_FER(x) - FER\_min\_casos)/(FER\_max\_casos-FER\_min\_casos)+(fun\_FER(x)>FER\_max\_casos).\*1; %maximizar

desej\_NEB=@(x) (fun\_NEB(x)<NEB\_min\_casos).\*0+ (fun\_NEB(x)>=NEB\_min\_casos & fun\_NEB(x)<=NEB\_max\_casos).\*(fun\_NEB(x) - NEB\_min\_casos)/(NEB\_max\_casos-NEB\_min\_casos)+(fun\_NEB(x)>NEB\_max\_casos).\*1; %maximizar

desej\_FEB=@(x) (fun\_FEB(x)<FEB\_min\_casos).\*0+ (fun\_FEB(x)>=FEB\_min\_casos & fun\_FEB(x)<=FEB\_max\_casos).\*(fun\_FEB(x) - FEB\_min\_casos)/(FEB\_max\_casos-FEB\_min\_casos)+(fun\_FEB(x)>FEB\_max\_casos).\*1; %maximizar

desej\_Ex=@(x) (fun\_Ex(x)<Ex\_min\_casos).\*0+ (fun\_Ex(x)>=Ex\_min\_casos & fun\_Ex(x)<=Ex\_max\_casos).\*(fun\_Ex(x) - Ex\_min\_casos)/(Ex\_max\_casos-Ex\_min\_casos)+(fun\_Ex(x)>Ex\_max\_casos).\*1; %maximizar

desej\_IC=@(x) (fun\_IC(x)<0.9\*IC\_min\_casos).\*1+ (fun\_IC(x)>=0.9\*IC\_min\_casos & fun\_IC(x)<=IC\_max\_casos/1.1).\*(fun\_IC(x) - IC\_max\_casos/1.1)/(0.9\*IC\_min\_casos-IC\_max\_casos/1.1)+(fun\_IC(x)>IC\_max\_casos/1.1).\*0; %minimizar

desej\_AbtCO2=@(x) (fun\_AbtCO2(x)<AbtCO2\_min\_casos).\*0+ (fun\_AbtCO2(x)>=AbtCO2\_min\_casos & fun\_AbtCO2(x)<=AbtCO2\_max\_casos).\*(fun\_AbtCO2(x) - AbtCO2\_min\_casos)/(AbtCO2\_max\_casos-AbtCO2\_min\_casos)+(fun\_AbtCO2(x)>AbtCO2\_max\_casos).\*1; %maximizar

desej\_VPL=@(x) (fun\_VPL(x)<0).\*0+ (fun\_VPL(x)>=0 & fun\_VPL(x)<=VPL\_max\_casos).\*(fun\_VPL(x) - 0)/(VPL\_max\_casos-0)+(fun\_VPL(x)>VPL\_max\_casos).\*1; %maximizar

desej\_TIR=@(x) (fun\_TIR(x)<0.1).\*0+ (fun\_TIR(x)>=0.1 & fun\_TIR(x)<=TIR\_max\_casos).\*(fun\_TIR(x) - 0.1)/(TIR\_max\_casos-0.1)+(fun\_TIR(x)>TIR\_max\_casos).\*1; %maximizar

%Função desejabilidade global considerando o mesmo peso para cada classe de

%resposta. NER e NEB = classe de energia; FER e FEB = classe de

%sustentabildiade; Ex = classe de exergia; IC e AbtCO2 = classe de

%emissoes; VPL e TIR = classe de economia

peso\_unifporclasse\_NER = 2/3;

peso\_unifporclasse\_FER = 0;

peso\_unifporclasse\_NEB = 2/3;

peso\_unifporclasse\_FEB = 2/3;

peso\_unifporclasse\_Ex = 2;

peso\_unifporclasse\_IC = 1;

peso\_unifporclasse\_AbtCO2 = 1;

peso\_unifporclasse\_VPL = 1;

peso\_unifporclasse\_TIR = 1;

soma\_pesos\_unifporclasse = peso\_unifporclasse\_NER + peso\_unifporclasse\_FER + peso\_unifporclasse\_NEB + peso\_unifporclasse\_FEB + peso\_unifporclasse\_Ex + peso\_unifporclasse\_IC + peso\_unifporclasse\_AbtCO2 + peso\_unifporclasse\_VPL + peso\_unifporclasse\_TIR;

desej\_global\_unifporclasse = @(x) (desej\_NER(x)^peso\_unifporclasse\_NER\*desej\_FER(x)^peso\_unifporclasse\_FER\*desej\_NEB(x)^peso\_unifporclasse\_NEB\*desej\_FEB(x)^peso\_unifporclasse\_FEB\*desej\_Ex(x)^peso\_unifporclasse\_Ex\*desej\_IC(x)^peso\_unifporclasse\_IC\*desej\_AbtCO2(x)^peso\_unifporclasse\_AbtCO2\*desej\_VPL(x)^peso\_unifporclasse\_VPL\*desej\_TIR(x)^peso\_unifporclasse\_TIR)^(1/soma\_pesos\_unifporclasse);

%Funcao desejabilidade global com pesos subjetivos

peso\_subj\_NER = (3.25+3.81)/100;

peso\_subj\_FER = 0; %3.81/100;

peso\_subj\_NEB = 5.19/100;

peso\_subj\_FEB = 5.25/100;

peso\_subj\_Ex = 16.25/100;

peso\_subj\_IC = 10.75/100;

peso\_subj\_AbtCO2 = 21.75/100;

peso\_subj\_VPL = 15.25/100;

peso\_subj\_TIR = 18.50/100;

soma\_pesos\_subj = peso\_subj\_NER + peso\_subj\_FER + peso\_subj\_NEB + peso\_subj\_FEB + peso\_subj\_Ex + peso\_subj\_IC + peso\_subj\_AbtCO2 + peso\_subj\_VPL + peso\_subj\_TIR;

desej\_global\_subj = @(x) (desej\_NER(x)^peso\_subj\_NER\*desej\_FER(x)^peso\_subj\_FER\*desej\_NEB(x)^peso\_subj\_NEB\*desej\_FEB(x)^peso\_subj\_FEB\*desej\_Ex(x)^peso\_subj\_Ex\*desej\_IC(x)^peso\_subj\_IC\*desej\_AbtCO2(x)^peso\_subj\_AbtCO2\*desej\_VPL(x)^peso\_subj\_VPL\*desej\_TIR(x)^peso\_subj\_TIR)^(1/soma\_pesos\_subj);

%Maximização das funções desejabilidades

fun\_desej\_global\_unifporclasse = @(x) -desej\_global\_unifporclasse(x);

fun\_desej\_global\_subj = @(x) -desej\_global\_subj(x);

x\_max\_desej\_global\_unifporclasse = fmincon(fun\_desej\_global\_unifporclasse, x0,A,b,Aeq,beq,lb,ub,nonlcon); desej\_global\_unifporclasse\_max = desej\_global\_unifporclasse(x\_max\_desej\_global\_unifporclasse);

x\_max\_desej\_global\_subj = fmincon(fun\_desej\_global\_subj, x0,A,b,Aeq,beq,lb,ub,nonlcon); desej\_global\_subj\_max = desej\_global\_subj(x\_max\_desej\_global\_subj);

%Matriz de ptos otimos. A matriz x tem as coordenadas para cada resposta nas

%linhas, ex. a x otima da NER esta na linha 1, x otima TIR ta na linha 9

Matriz\_x\_otimos = [x\_max\_NER

x\_max\_FER

x\_max\_NEB

x\_max\_FEB

x\_max\_Ex

x\_max\_IC

x\_max\_AbtCO2

x\_max\_VPL

x\_max\_TIR];

%Matriz coluna com as respostas otimas encontradas

Matriz\_Respostas\_otimos = [NER\_max

FER\_max

NEB\_max

FEB\_max

Ex\_max

IC\_min

AbtCO2\_max

VPL\_max

TIR\_max];

%Matriz com desejabildiades otimas. Desejabilidade uniforme, Desejabilidade

%uniforme para as classes, e adesejabilidade subjetiva

Matriz\_x\_desej\_globais\_otimas = [x\_max\_desej\_global\_unifporclasse

x\_max\_desej\_global\_subj];

Matriz\_desej\_globais\_otimas=[desej\_global\_unifporclasse\_max

desej\_global\_subj\_max];

Axcel = [Matriz\_desej\_globais\_otimas Matriz\_x\_desej\_globais\_otimas] %desej globais e coordendas otimas para os dois critérios de pesos

%Reporta o valor dos indicadores e a desejabilidade indiv. no pto xis

xis = x\_max\_desej\_global\_subj;

funcoes\_e\_xises = [

fun\_NER(xis) desej\_NER(xis)

fun\_FER(xis) desej\_FER(xis)

fun\_NEB(xis) desej\_NEB(xis)

fun\_FEB(xis) desej\_FEB(xis)

fun\_Ex(xis) desej\_Ex(xis)

fun\_IC(xis) desej\_IC(xis)

fun\_AbtCO2(xis) desej\_AbtCO2(xis)

fun\_VPL(xis) desej\_VPL(xis)

fun\_TIR(xis) desej\_TIR(xis)];

%Area de plotagem para X3 = 0

% plot\_desej\_global\_uniforme=zeros(100,100);

x1\_0 = -1; x1\_next = x1\_0;

x2\_0 = -1; x2\_next = x2\_0;

dx = 0.02;

for i = 1:101

x1\_next=x1\_0+(i-1)\*dx;

for j = 1:101

x2\_next=x2\_0+(j-1)\*dx;

plot\_desej\_global\_unifporclasse(i,j) = desej\_global\_unifporclasse([x1\_next x2\_next 0]);

end

end

for i = 1:101

x1\_next=x1\_0+(i-1)\*dx;

for j = 1:101

x2\_next=x2\_0+(j-1)\*dx;

plot\_desej\_global\_subj(i,j) = desej\_global\_subj([x1\_next x2\_next 0]);

end

end

figure(1)

subplot(1,2,1)

[X,Y]=meshgrid(-1:0.02:1);

surf(X,Y,plot\_desej\_global\_unifporclasse,'edgecolor','none')

xlabel('Ef. Biofix. (X\_2)')

ylabel('Bag. p. Hidr. (X\_1)')

title('Desej. global com pesos iguais para as categorias')

colorbar;

subplot(1,2,2)

[X,Y]=meshgrid(-1:0.02:1);

surf(X,Y,plot\_desej\_global\_subj,'edgecolor','none')

xlabel('Ef. Biofix. (X\_2)')

ylabel('Bag. p. Hidr. (X\_1)')

title('Desej. global com hierarquia de pesos')

colorbar;

%Area de plotagem para X2 = 0

x1\_0 = -1; x1\_next = x1\_0;

x2\_0 = -1; x2\_next = x2\_0;

dx = 0.02;

for i = 1:101

x1\_next=x1\_0+(i-1)\*dx;

for j = 1:101

x2\_next=x2\_0+(j-1)\*dx;

plot\_desej\_global\_unifporclasse(i,j) = desej\_global\_unifporclasse([x1\_next 0 x2\_next]);

end

end

for i = 1:101

x1\_next=x1\_0+(i-1)\*dx;

for j = 1:101

x2\_next=x2\_0+(j-1)\*dx;

plot\_desej\_global\_subj(i,j) = desej\_global\_subj([x1\_next 0 x2\_next]);

end

end

figure(2)

subplot(1,2,1)

[X,Y]=meshgrid(-1:0.02:1);

surf(X,Y,plot\_desej\_global\_unifporclasse,'edgecolor','none')

xlabel('Teor de lipídeos (X\_3)')

ylabel('Bag. p. Hidr. (X\_1)')

title('Desej. global com pesos iguais para as categorias')

colorbar;

subplot(1,2,2)

[X,Y]=meshgrid(-1:0.02:1);

surf(X,Y,plot\_desej\_global\_subj,'edgecolor','none')

xlabel('Teor de lipídeos (X\_3)')

ylabel('Bag. p. Hidr. (X\_1)')

title('Desej. global com hierarquia de pesos')

colorbar;