DECKBLATT

# Aufgabenstellung

Ein Stoffgemisch aus Xylol, Styrol, Toluol und Benzol wird über 3 Rektifikationskolonne aufgetrennt (Abbildung 1). In dieser Übung soll am Beispiel der angegebenen Massenprozente die einzelnen Massenströme über ein lineares Gleichungssystem (LGS) gelöst werden. Hierfür wird ein genereller MATLAB Code mit Hilfe der Gauss-Seidel Methode in MATLAB programmiert werden.

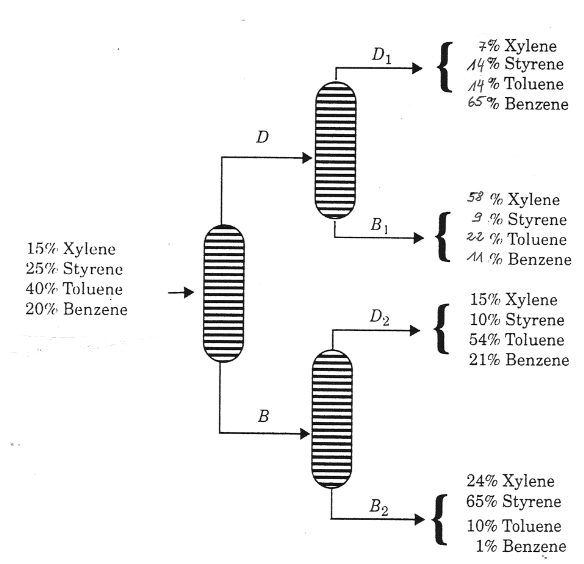


Abbildung 1: Prozessdarstellung der Rektifikationskolonnen

# Auswertung

Herleitung des LGS

Massenbilanz:

Komponentenbilanz:

mit

Daraus resultiert das LGS in A x= b Schreibweise:

Programmierung in Matlab

%Gauss Seidel Solver

%from source https://de.mathworks.com/matlabcentral/answers/

%303634-writing-a-code-to-attempt-to-

%make-matrix-diagonally-dominant 2017-11-05

clear;

clc;

%Input square Matrix of A

A= [

.7 .58 .15 .24;

.14 .09 .1 .65;

.14 .22 .54 .10;

.65 .11 .21 .01];

%Input constant vector

b= [.15 ; .25; .4; .2];

[m,n]=size(A);

%checks if Matrix is square

if m~=n, error('A is not square matrix');

end

ww=1;

%Rearrange Matrix in diagonal dominant order

while ww <= n

for i=1:n

if sum(abs(A(i,:))) - abs(A(i,ww)) < abs(A(i,ww))

A([ww,i],:)=A([i ww],:);

b([ww i],:)=b([i ww],:);

ww=ww+1;

break

elseif sum(abs(A(i,:))) - abs(A(i,ww)) == abs(A(i,ww))

A([ww,i],:)=A([i ww],:);

b([ww i],:)=b([i ww],:);

if sum(abs(A(i,:))) - abs(A(i,ww)) < abs(A(i,ww))

A([ww,i],:)=A([i ww],:);

b([ww i],:)=b([i ww],:);

ww=ww+1;

break;

end

if i==n

ww=ww+1;

end

elseif i==n

ww=ww+1;

else

continue

end

end

end

fprintf ('Matrix A successfully rearranged to diagonal dominat')

disp(A);

% Iterations Gauss-Seidel

% Input Tolernace (TOL), MaxNumOfIter, initial x-value,

TOL=1e-4;

MaxNumOfIter=1000;

x=zeros(n,1);

counter=1;

iteration = 0;

Error\_eval= ones(1,n);

while max(Error\_eval) > 0.001

iteration = iteration + 1;

Z = x; % save current values to calculate error later

for i = 1:n

j = 1:n; % define an array of the coefficients' elements

j(i) = []; % eliminate the unknow's coefficient from the remaining coefficients

Xtemp = x; % copy the unknows to a new variable

Xtemp(i) = []; % eliminate the unknown under question from the set of values

x(i) = (b(i) - sum(A(i,j) \* Xtemp)) / A(i,i);

end

Xsolution(:,iteration) = x;

Error\_eval = sqrt((x - Z).^2);

end

%% Display Results

fprintf ('The soltuion of x is: \n %f\n %f\n %f\n', x)

fprintf('number of iterations is: %f\n', iteration)