processes

MDPI

Article

A Quality Integrated Fuzzy Inference System for the Reliability Estimating of Fluorochemical Engineering Processes

Feng Xue [1], Xintong Li [1], Kun Zhou [1], Xiaoxia Ge [2], Weiping Deng [2], Xu Chen [1] and Kai Song [1,*]

[1] School of Chemical Engineering and Technology, Tianjin University, Tianjin 300350, China; 3014207238@tju.edu.cn (F.X.); 3014207218@tju.edu.cn (X.L.); kzhou@tju.edu.cn (K.Z.); xchen@tju.edu.cn (X.C.)

[2] Health, Safety and Environmental Protection Department, Juhua Group Co., Ltd., Quzhou 324004, China; ahbgxx@juhua.com.cn (X.G.); jhdwp@juhua.com.cn (W.D.)

* Correspondence: ksong@tju.edu.cn; Tel.: +86-189-2031-7821

Citation: Xue, F.; Li, X.; Zhou, K.; Ge, X.; Deng, W.; Chen, X.; Song, K. A Quality Integrated Fuzzy Inference System for the Reliability Estimating of Fluorochemical Engineering Processes. *Processes* **2021**, *9*, 292. https://doi.org/10.3390/pr9020292

Academic Editor: Jie Zhang and Meihong Wang

Received: 11 January 2021

Accepted: 31 January 2021

Published: date

MATLAB codes for TEP

```
close all;
clear all;
clc;
load('plsdata.mat');
for i=1:5
    % Get training and test data
    testx=plstrainx(192*i-191:192*i,:);
    testy=plstrainy(192*i-191:192*i,:);
    trainx=plstrainx;
    trainx(192*i-191:192*i,:)=[];
    trainx;
    trainy=plstrainy;
    trainy(192*i-191:192*i,:)=[];
    trainy;
    % Standardized training data and testing data
    [trstx,trsty,trsave]=pretreat(trainx,trainy);
    Junsave(i,:)=trsave(1,:);
    Fangsave(i,:)=trsave(2,:);
    testy1=testy(:,1)./testy(:,2);
    hebi=[testx,testy1];
    for t=1:192
        jun(t,:)=trsave(1,:)
        fang(t,:)=trsave(2,:)
    end
    hebi2=hebi-jun;
    hebi3=hebi2./fang
    teststx=hebi3(:,1:32);
    teststy=hebi3(:,33);

    % PLS for training data
    for ncomp=1:32
        [xl,yl,xs,ys,beta,pctvar,mse,stats]=plsregress(trstx,trsty,ncomp);
        K=ones(768,1);
        X3=[K,trstx];
        Y0=X3*beta;
        DT=trsty-Y0;
        DT1=DT*trsave(2,33);
        b=sum(DT1.*DT1,1);
        trRMSE=sqrt(b/768);
        trrmse(ncomp,:)=trRMSE;
        beta1(ncomp,:)=beta;
        K1=ones(192,1);
        x3=[K1,teststx];
        Y1=x3*beta;
        DT1=teststy-Y1;
        DT1=DT1*trsave(2,33);
        b1=sum(DT1.*DT1,1);
        teRMSE=sqrt(b1/192);
        termse(ncomp,:)=teRMSE;

        % RMSE for testing data
Junsave1=mean(Junsave,1);
Fangsave1=mean(Fangsave,1);
```

```
plstesty1=plstesty(:,1)./plstesty(:,2);
heb=[plstestx,plstesty1];
    for t=1:240
        jun1(t,:)=Junsave1;
        fang1(t,:)=Fangsave1;
    end
    heb2=heb-jun1;
    heb3=heb2./fang1;
    plsteststx=heb3(:,1:32);
    plsteststy=heb3(:,33);

    K2=ones(240,1);
        x3=[K2,plsteststx];
        Y2=x3*beta;
        DT=plsteststy-Y2;
        DT1=DT*trsave(2,33);
        b2=sum(DT1.*DT1,1);
        tteRMSE=sqrt(b2/240);
        ttermse(ncomp,:)=tteRMSE;
    end
    TRrmse(:,i)=trrmse;
    TErmse(:,i)=termse;
    TTErmse(:,i)=ttermse;
    BETA(:,i*33-32:i*33)=beta1;
end
% RMSE
TRrmse1=mean(TRrmse,2);
TErmse1=mean(TErmse,2);
TTErmse1=mean(TTErmse,2);
 %  Calculation for Beta
 nc=1;
 BetA=BETA(nc,:);
 BetA=[BetA(:,1:33);BetA(:,34:66);BetA(:,67:99);BetA(:,100:132);BetA(:,133:165)];
 tbeta=mean(BetA,1);
tbeta1=abs(tbeta);
tbeta2=tbeta1/sum(tbeta1);

% System reliability estimation for fault 6 observations
load('t6.mat');
simout1=simout(1:120,:);
GHbi=simout1(:,40)./simout1(:,41);
xmv(:,9)=[];
xmv(:,11)=[];
xmv1=xmv(1:120,:);
PLSdata=[simout1(:,1:22),xmv1];
load('junfang.mat');
    for t=1:120
        jun1(t,:)=Junsave1;
        fang1(t,:)=Fangsave1;
    end
    PLSdata1=PLSdata-jun1;
    PLSdata2=PLSdata1./fang1;
%Quality loss for fault 6 observations
tbeta1=abs(tbeta);
```

```
tbeta2=tbeta1/sum(tbeta1);
beta=tbeta2(:,2:33);
for m=1:120
      betaa(m,:)=beta
end
qloss0=PLSdata2.*betaa;
xishu=-0.5*qloss0.*qloss0;
e=exp(xishu);
qloss1=1-e;
qloss=sum(qloss1,2);
% Obtain vital safety variables input
d1=(simout1(:,7)-2705)/190;
d2=(simout1(:,8)-50)/50;
d3=(simout1(:,9)-94.8)/55.2;
d4=(simout1(:,12)-30)/70;
d5=(simout1(:,15)-30)/70;
safedata=[d1,d2,d3,d4,d5];
fisdata=[safedata,qloss/12]
fisdata=abs(fisdata)
% Estimating system reliability by QFIS
fisdata(:,3)=[];
[System]
Name='TEFIS'
Type='mamdani'
Version=2.0
NumInputs=5
NumOutputs=1
NumRules=72
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='Reactorpressure'
Range=[0 1]
NumMFs=2
MF1='not-high':'trapmf',[-1.0 0 0.6 1.0]
MF2='high':'trimf',[0.5 1 1.5]

[Input2]
Name='Reactorlevel'
Range=[0 1]
NumMFs=3
MF1='low':'trimf',[0 0 0.3]
MF2='medium':'trimf',[0 0.35 0.62]
MF3='high':'trimf',[0.45 1 1]

[Input3]
Name='Procuctseperatorlevel'
Range=[0 1]
NumMFs=2
MF1='not-high':'trapmf',[-1.0 0 0.5 1.0]
```

MF2='high':'trimf',[0.6 1 1.5]

[Input4]
Name='Stripperbaselevel'
Range=[0 1]
NumMFs=3
MF1='low':'trimf',[0 0 0.5]
MF2='middle':'trimf',[0.3 0.6 0.9]
MF3='high':'trimf',[0.7 1 1]

[Input5]
Name='Qualityloss'
Range=[0 1]
NumMFs=2
MF1='low':'trimf',[-1 0 0.3]
MF2='not-low':'trapmf',[0.2 0.4 1 2]

[Output1]
Name='Reliability'
Range=[0 1]
NumMFs=4
MF1='danger':'trimf',[0 0 0.3]
MF2='alarm':'trimf',[0.1 0.35 0.6]
MF3='warning':'trimf',[0.4 0.65 0.9]
MF4='safe':'trimf',[0.7 1 1]

[Rules]
1 2 2 2 1, 3 (1) : 1
2 2 2 2 1, 2 (1) : 1
1 1 2 2 1, 2 (1) : 1
1 3 2 2 1, 2 (1) : 1
1 2 1 2 1, 2 (1) : 1
1 2 2 1 1, 2 (1) : 1
1 2 2 2 2, 2 (1) : 1
2 1 2 2 1, 2 (1) : 1
2 3 2 2 1, 2 (1) : 1
2 2 1 2 1, 2 (1) : 1
2 2 3 2 1, 2 (1) : 1
2 2 2 1 1, 2 (1) : 1
2 2 2 2 2, 2 (1) : 1
1 1 1 2 1, 2 (1) : 1
1 1 3 2 1, 2 (1) : 1
1 1 2 1 1, 2 (1) : 1
1 1 2 2 2, 2 (1) : 1
1 3 1 2 1, 2 (1) : 1
1 3 2 1 1, 2 (1) : 1
1 3 2 3 1, 2 (1) : 1
1 3 2 2 2, 2 (1) : 1
1 2 1 1 1, 2 (1) : 1
1 2 1 3 1, 2 (1) : 1
1 2 1 2 2, 2 (1) : 1
1 2 2 3 1, 2 (1) : 1
1 2 2 2 2, 2 (1) : 1
1 2 2 1 2, 2 (1) : 1

```
1 2 2 3 2, 2 (1) : 1
2 1 1 2 1, 4 (1) : 1
2 3 1 2 1, 4 (1) : 1
2 1 2 1 1, 4 (1) : 1
2 1 2 3 1, 4 (1) : 1
2 3 2 1 1, 4 (1) : 1
2 3 2 3 1, 4 (1) : 1
2 1 2 2 2, 4 (1) : 1
2 3 2 2 2, 4 (1) : 1
2 2 1 1 1, 4 (1) : 1
2 2 1 3 1, 4 (1) : 1
2 2 1 2 2, 4 (1) : 1
2 2 2 1 2, 4 (1) : 1
2 2 2 3 2, 4 (1) : 1
2 1 1 1 1, 4 (1) : 1
2 1 1 3 1, 4 (1) : 1
2 1 2 3 1, 4 (1) : 1
2 3 1 1 1, 4 (1) : 1
2 3 1 3 1, 4 (1) : 1
2 1 1 2 2, 4 (1) : 1
2 3 1 2 2, 4 (1) : 1
2 2 1 1 2, 4 (1) : 1
2 2 1 3 2, 4 (1) : 1
2 1 1 1 1, 4 (1) : 1
2 1 1 3 1, 4 (1) : 1
2 3 1 1 1, 4 (1) : 1
2 3 1 3 1, 4 (1) : 1
2 1 1 2 2, 4 (1) : 1
2 3 1 2 2, 4 (1) : 1
2 1 2 1 2, 4 (1) : 1
2 1 2 3 2, 4 (1) : 1
2 3 2 1 2, 4 (1) : 1
2 3 2 3 2, 4 (1) : 1
2 2 1 1 2, 4 (1) : 1
2 2 1 3 2, 4 (1) : 1
1 1 1 1 2, 4 (1) : 1
1 1 1 3 2, 4 (1) : 1
1 3 1 1 2, 4 (1) : 1
1 3 1 3 2, 4 (1) : 1
2 1 1 1 2, 1 (1) : 1
2 1 1 3 2, 1 (1) : 1
2 1 2 1 2, 1 (1) : 1
2 1 2 3 2, 1 (1) : 1
2 3 1 1 2, 1 (1) : 1
2 3 1 3 2, 1 (1) : 1
fismat = readfis('TEFIS');
output= evalfis(fisdata,fismat);
```