

بر اساس پروتکل‌های دوره‌های آموزشی آپتیم‌یار، به اشتراک‌گذاری محتوا و کدهای نرم‌افزاری منظر حقوقی ممنوع است و از منظر اخلاقی نارضایتی مدرس دوره و گروه آپتیم‌یار را به همراه دارد.

از توجه شما به پروتکل دوره‌های آموزشی آپتیم‌یار سپاسگزاریم.

دوره جامع آنلاین بهینه‌سازی استوار و برنامه‌ریزی در شرایط عدم قطعیت همراه با کدنویسی در نرم‌افزار (GAMS)

Decision-Making under Uncertainty (Robust Optimization - Stochastic Programming - Fuzzy Programming)

مدرس:

دکتر علی پاپی (Ali Papi)

تخصص شاخص: بهینه‌سازی و تحقیق در عملیات، علم تحلیل داده، تکنیک‌های تجزیه و روش‌های حل دقیق، بهینه‌سازی استوار داده‌محور، هوش محاسباتی و الگوریتم‌های فراابتکاری، نظریه بازی، بهینه‌سازی چندهدفه و تصمیم‌گیری چندمعیاره

Optimization & Operations Research, Data Analytics, Computational Intelligence & Metaheuristics, Decomposition Techniques & Exact Methods, Data-Driven Robust Optimization, Game Theory, Multi Criteria Decision Making

GeneralModel_BOX

GeneralModel_nominal

GeneralModel Validation

GeneralModel Validation moreCriteria



اخطار: بر اساس پروتکل‌های دوره‌های آموزشی آپتیم‌یار، به اشتراک‌گذاری محتوا و کدهای نرم‌افزاری منظر حقوقی ممنوع است و از منظر اخلاقی نارضایتی مدرس دوره و گروه آپتیم‌یار را به همراه دارد.

باز توجه شما به پروتکل دوره‌های آموزشی آپتیم‌یار بسیار سپاسگزاریم.

GeneralModel BOX

Sets

i /i1*i20/

j /j1*j100/

;

Parameters

c(j)

a_N(i,j)

PR

b(i)

;

c(j)=uniform(10,30);

a_N(i,j)=uniform(5,10);

b(i)= uniform(100,200);

PR=0.3;

*** BOX (p = infinity)

Scalar say;

say=1;



OptimYar

Positive Variable

x(j)

;

Free Variable

z

;

Equations

obj

cons

;

obj.. z =e= sum(j,c(j)*x(j));

cons(i).. sum(j,a_N(i,j)*x(j)) - say*sum(j,abs(PR*a_N(i,j))*x(j)) =g= b(i);

;

Model GeneralModel

/

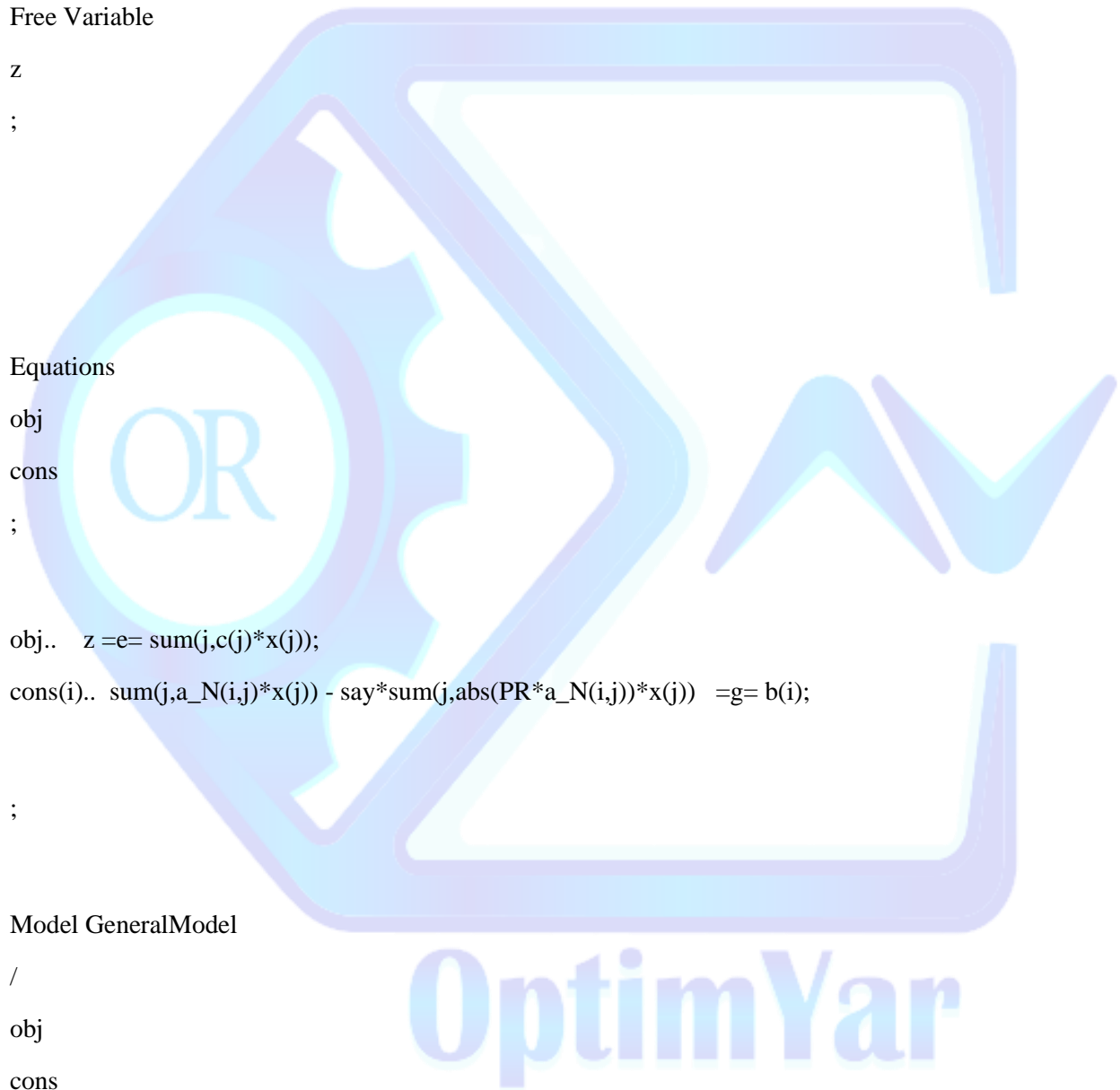
obj

cons

/

Options

LP = CPLEX



```
reslim =100
```

```
optcr =0
```

```
;
```

```
set iter /iter1*iter11/;
```

```
Parameter Result(iter,*);
```

```
;
```

```
Loop(iter,
```

OR

```
say = (ord(iter)-1)/(card(iter)-1);
```

```
Solve GeneralModel us LP min z ;
```

```
Result(iter,'say')=say;
```

```
Result(iter,'cost')=z.l;
```

```
display
```

```
say
```

```
x.l
```

```
z.l
```

```
;
```

OptimYar

);

Display Result



GeneralModel nominal

Sets

i /i1*i20/

j /j1*j100/

;

Parameters

c(j)

a_N(i,j)

PR

b(i)

;

c(j)=uniform(10,30);

a_N(i,j)=uniform(5,10);

b(i)= uniform(100,200);

PR=0.3;

Positive Variable

x(j)

;

Free Variable

z



;

Equations

obj

cons

;

obj.. z =e= sum(j,c(j)*x(j));

cons(i).. sum(j,a_N(i,j)*x(j)) =g= b(i);

;

Model GeneralModel

/

obj

cons

/

Options

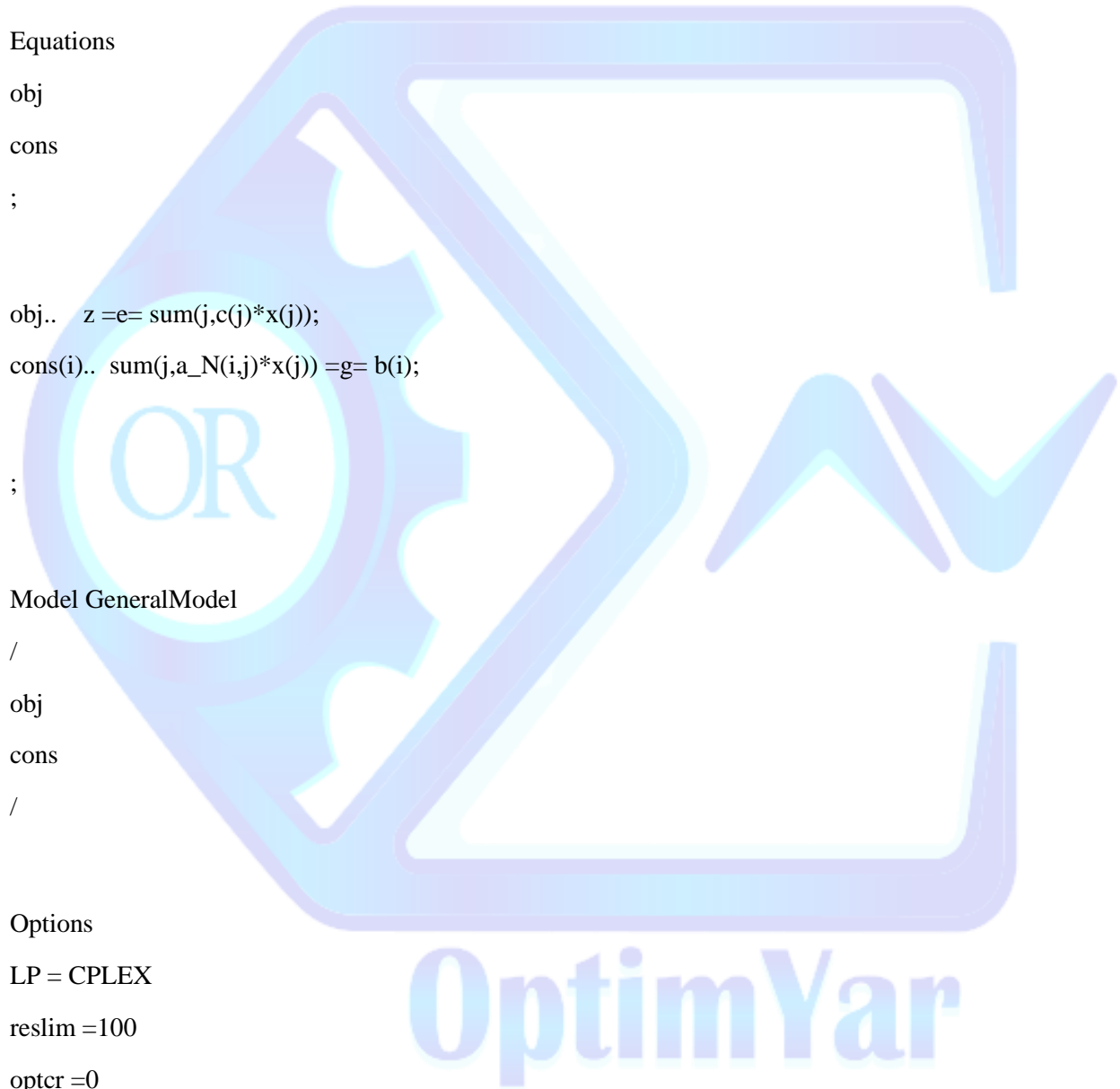
LP = CPLEX

reslim =100

optcr =0

;

Solve GeneralModel us LP min z



display

x.l

z.l

;



GeneralModel Validation

Sets

i /i1*i20/

j /j1*j100/

;

Parameters

c(j)

a_N(i,j)

PR

b(i)

;

c(j)=uniform(10,30);

a_N(i,j)=uniform(5,10);

b(i)= uniform(100,200);

PR=0.05;

*** BOX (p = infinity)

Scalar say;

say=1;

Positive Variable

x(j)

;



OptimYar

Free Variable

z
;

Equations

obj

cons

;

obj.. z =e= sum(j,c(j)*x(j));

cons(i).. sum(j,a_N(i,j)*x(j)) - say*sum(j,abs(PR*a_N(i,j))*x(j)) =g= b(i);

;

*****Realization

set iter /1*10/;

OptimYar

Parameter

Result(iter,*)

Parameters

a_R(i,j)

;

Scalar BestZ;

Positive Variable s(i);

;

Equations

obj_R

cons_R

;

parameter

PC(i)

;

PC(i)= uniform(2,5);

obj_R.. z =e= sum(j,c(j)*x(j)) + sum(i,pc(i)*s(i)) ;



cons_R(i).. sum(j,a_R(i,j)*x(j)) =g= b(i) - s(i);

Model GeneralModel

/

obj

cons

/

Model GeneralModel_Realized

/

obj_R

cons_R

/

Options

LP = CPLEX

reslim =100

optcr =0

;

**Criteria



Parameters

ROUR(iter)

*Relative Optimality Un-Robustness (ROUR)

M_ROUR

A_ROUR

;

Loop(iter,

$a_{R(i,j)} = \text{uniform}((a_{N(i,j)} - \text{abs}(PR*a_{N(i,j)})) + (\text{ord}(\text{iter})-1)/\text{card}(\text{iter})*\text{abs}(PR*a_{N(i,j)}), (a_{N(i,j)} - \text{abs}(PR*a_{N(i,j)})) + (\text{ord}(\text{iter}))/\text{card}(\text{iter})*\text{abs}(PR*a_{N(i,j)}))$);

**

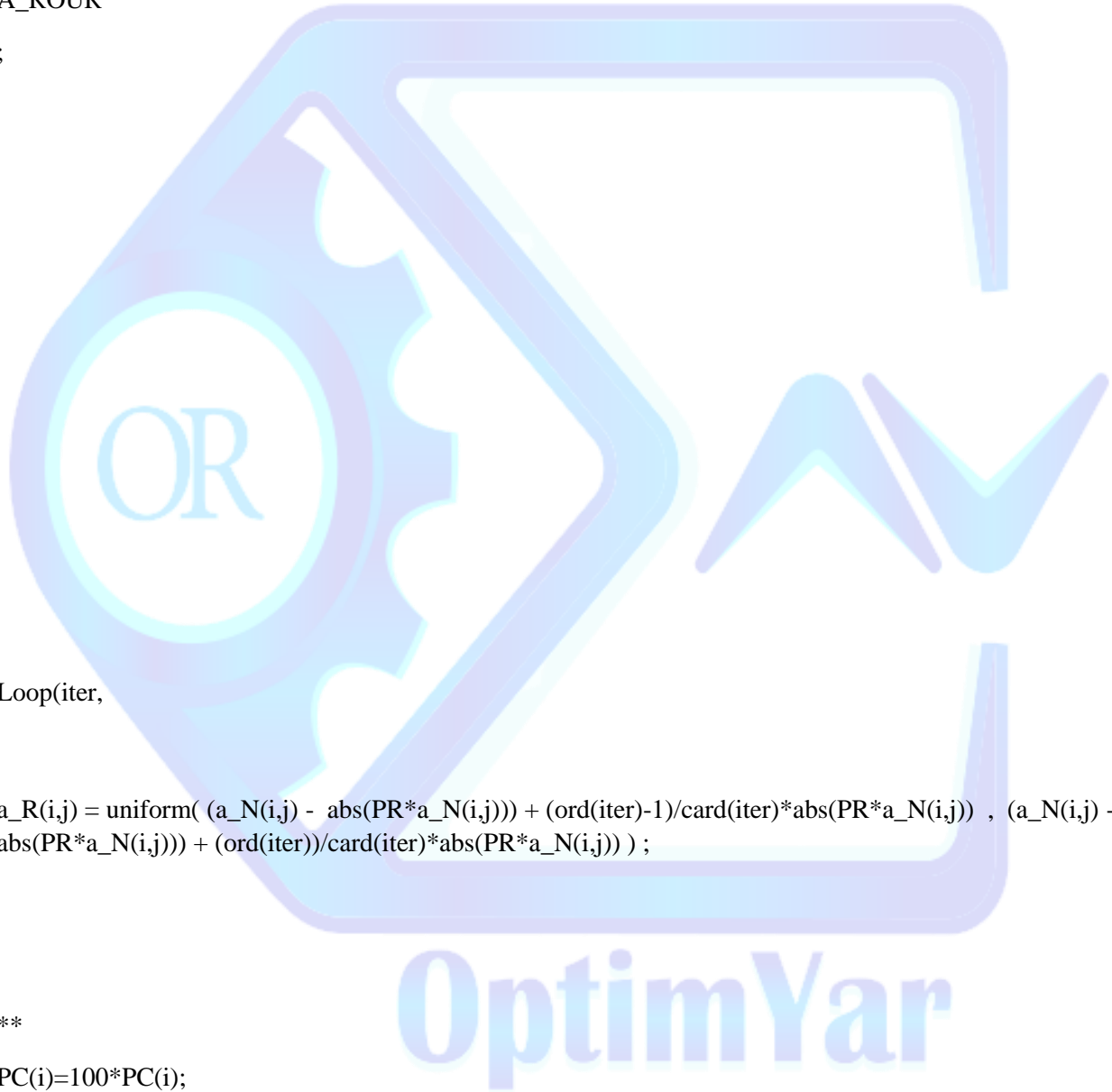
PC(i)=100*PC(i);

Solve GeneralModel_Realized us LP min z ;

BestZ=z.l;

Display

"Best Solution"



```
z.l
```

```
s.l
```

```
*s should be zero
```

```
;
```

```
Result(iter,"BestZ") = z.l ;
```

```
**
```

```
**
```

```
Solve GeneralModel us LP min z ;
```

```
Display
```

```
"Robust"
```

```
x.l
```

```
;
```

```
**
```



OptimYar

```
**
```

```
PC(i)=1/100*PC(i);
```

```
x.fx(j)=x.l(j);
```

```
Solve GeneralModel_Realized us LP min z ;
```

```
Display
```

```
"Vadiation"
```

```
x.l
```

```
s.l
```

```
z.l
```

```
;
```

```
Result(iter,"RobustZ") = z.l ;
```

```
ROUR(iter)= 100*(Result(iter,"RobustZ") - Result(iter,"BestZ")) /Result(iter,"BestZ");
```

```
Result(iter,"ROUR")=ROUR(iter);
```

```
**
```

```
x.lo(j)=0;
```

```
x.up(j)=inf;
```

```
);
```

```
M_ROUR = smax(iter,ROUR(iter));
```

The logo for OptimYar features a large, stylized blue gear with the letters 'OR' inside it. To the right of the gear is a blue wave-like shape. Below these elements, the word 'OptimYar' is written in a large, blue, sans-serif font.

OptimYar


```
A_ROUR = sum(iter,ROUR(iter))/card(iter);
```

```
Display
```

```
Result
```

```
M_ROUR
```

```
A_ROUR
```

```
;
```



General Model Validation more Criteria

Sets

$i / i1 * i20 /$

$j / j1 * j100 /$

;

Parameters

$c(j)$

$a_N(i,j)$

PR

$b(i)$

;

$c(j) = \text{uniform}(10,30);$

$a_N(i,j) = \text{uniform}(5,10);$

$b(i) = \text{uniform}(100,200);$

PR=0.1;

*** BOX (p = infinity)

Scalar say;

say=0.8;

Positive Variable

$x(j)$

;



OptimYar

Free Variable

z
;

Equations

obj

cons

;

obj.. z =e= sum(j,c(j)*x(j));

cons(i).. sum(j,a_N(i,j)*x(j)) - say*sum(j,abs(PR*a_N(i,j))*x(j)) =g= b(i);

;

*****Realization

set iter /1*10/;

OptimYar

Parameter

Result(iter,*)

Parameters

a_R(i,j)

;

Scalar BestZ;

Positive Variable s(i);

;

Equations

obj_R

cons_R

;

parameter

PC(i)

;

PC(i)= uniform(0.2,0.4);

obj_R.. z =e= sum(j,c(j)*x(j)) + sum(i,pc(i)*s(i)) ;



cons_R(i).. sum(j,a_R(i,j)*x(j)) =g= b(i) - s(i);

Model GeneralModel

/

obj

cons

/

Model GeneralModel_Realized

/

obj_R

cons_R

/

Options

LP = CPLEX

reslim =100

optcr =0

;

**Criteria



OptimYar

Parameters

ROUR(iter)

*Relative Optimality Un-Robustness (ROUR)

M_ROUR

A_ROUR

FR(iter)

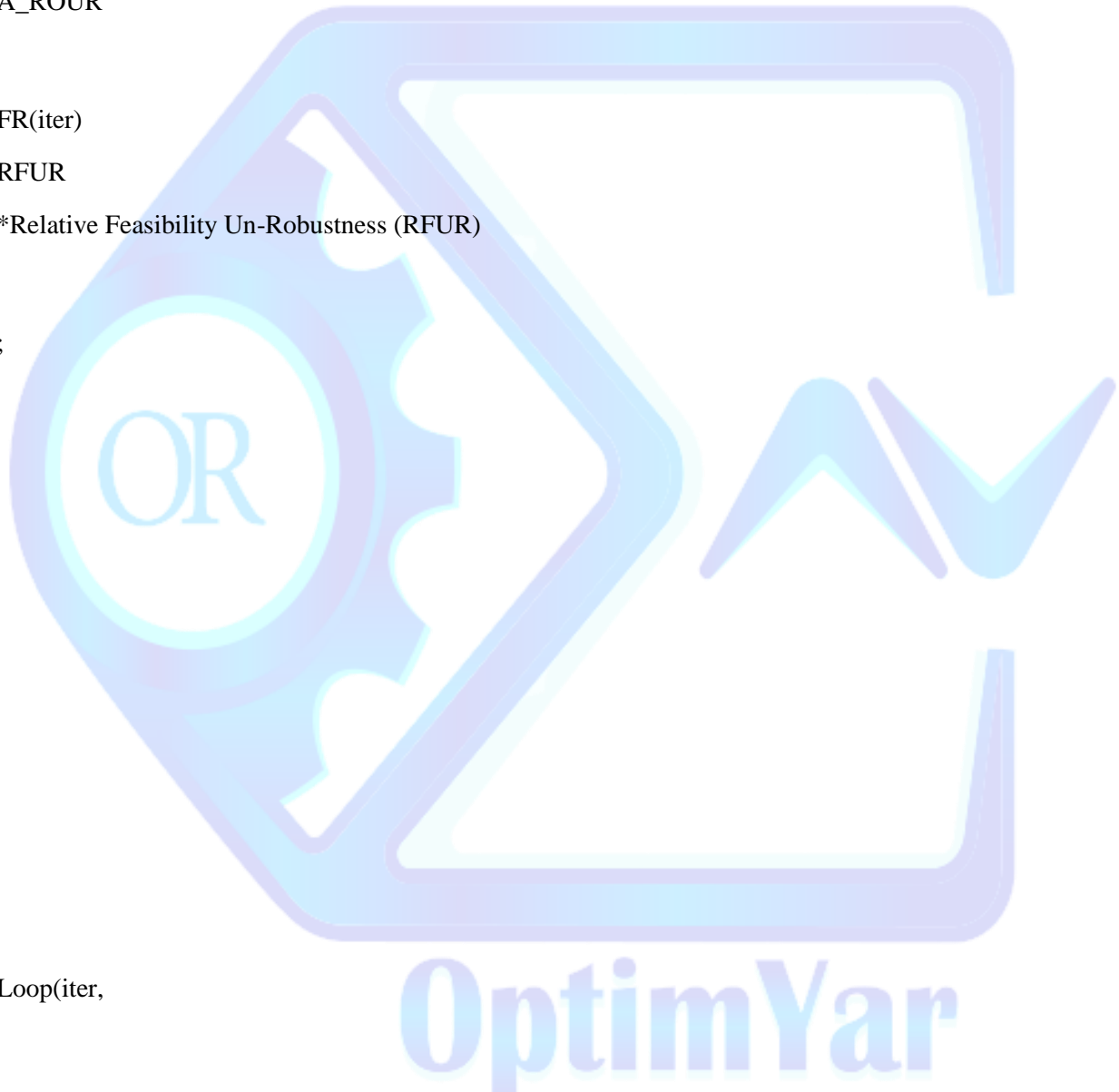
RFUR

*Relative Feasibility Un-Robustness (RFUR)

;

Loop(iter,

$a_R(i,j) = \text{uniform}((a_N(i,j) - \text{abs}(PR*a_N(i,j))) + (\text{ord}(\text{iter})-1)/\text{card}(\text{iter})*\text{abs}(PR*a_N(i,j)) , (a_N(i,j) - \text{abs}(PR*a_N(i,j))) + (\text{ord}(\text{iter}))/\text{card}(\text{iter})*\text{abs}(PR*a_N(i,j))) ;$



**

PC(i)=100*PC(i);

Solve GeneralModel_Realized us LP min z ;

BestZ=z.l;

Display

"Best Solution"

z.l

s.l

*s should be zero

;

Result(iter,"BestZ") = z.l ;

**

OR

**

Solve GeneralModel us LP min z ;

Display

"Robust"

x.l

;

**

OptimYar

**

PC(i)=1/100*PC(i);

x.fx(j)=x.l(j);

Solve GeneralModel_Realized us LP min z ;

Display

"Vadiation"

x.l

s.l

z.l

;

Result(iter,"RobustZ") = z.l ;

ROUR(iter)= 100*(Result(iter,"RobustZ") - Result(iter,"BestZ")) /Result(iter,"BestZ");

Result(iter,"ROUR")=ROUR(iter);

**

if (sum(i,s.l(i))<0.0001,

FR(iter)=1;

);


```
Result(iter,"FR")=FR(iter);
```

```
x.lo(j)=0;
```

```
x.up(j)=inf;
```

```
);
```

```
M_ROUR = smax(iter,ROUR(iter));
```

```
A_ROUR = sum(iter,ROUR(iter))/card(iter);
```

```
RFUR = sum(iter,1-FR(iter))/card(iter);
```

```
Display
```

```
Result
```

```
M_ROUR
```

```
A_ROUR
```

```
RFUR
```

```
;
```



OptimYar

دوره جامع آنلاین بهینه‌سازی استوار و برنامه‌ریزی در شرایط عدم قطعیت همراه با کدنویسی در نرم‌افزار (GAMS)

Decision-Making under Uncertainty (Robust Optimization - Stochastic Programming - Fuzzy Programming)

مدرس:

دکتر علی پاپی (Ali Papi)

تخصص شاخص: بهینه‌سازی و تحقیق در عملیات، علم تحلیل داده، تکنیک‌های تجزیه و روش‌های حل دقیق، بهینه‌سازی استوار داده‌محور، هوش محاسباتی و الگوریتم‌های فراابتکاری، نظریه بازی، بهینه‌سازی چندهدفه و تصمیم‌گیری چندمعیاره

Optimization & Operations Research, Data Analytics, Computational Intelligence & Metaheuristics, Decomposition Techniques & Exact Methods, Data-Driven Robust Optimization, Game Theory, Multi Criteria Decision Making