www.optimyar.com

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 عدم‌قطعیت 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 فرااِبتکاری، 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

کدهای نرم‌افزاری

SCND_BoxEllipsoidal

SCND_BoxPolyhedral

SCND_BoxPolyhedral_Case2

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 سپاسگزاریم.

SCND_BoxEllipsoidal

```
Sets
S       /s1*s10/
D       /d1*d20/
C       /c1*c30/
w       /w1*w5/
;


Parameters
A(s)
f(d)
b(s)
trSD_L(s,d)
trSD_R(s,d)
trSD_N(s,d)
trSD_P(s,d)
trDC(d,c)
p
dem(c,w)
dem_N(c)
capD(d)
capS(s)
Prob(w)
/
w1      0.2
w2      0.3
w3      0.1
w4      0.2
```

```
w5      0.2
/


;



A(s)    = uniform(1000,1500);
f(d)    = uniform(2000,3000);
b(s)    = uniform(5,10);
trSD_L(s,d)= uniform(1,1.5);
trSD_R(s,d)= uniform(2,3);

trSD_N(s,d)= (trSD_L(s,d)+trSD_R(s,d))/2;
trSD_P(s,d)= (trSD_R(s,d)-trSD_L(s,d))/2;

trDC(d,c)= uniform(0.5,0.7);
p       = 15;
capD(d)  = uniform(500,1000);
capS(s)  = uniform(1000,2000);



dem(c,'w1')   = uniform(50,100);
dem(c,'w2')   = (1+0.8)*dem(c,'w1') ;
dem(c,'w3')   = (1+0.2)*dem(c,'w1') ;
dem(c,'w4')   = (1-0.2)*dem(c,'w1') ;
dem(c,'w5')   = (1-0.3)*dem(c,'w1') ;

dem_N(c)= sum(w,prob(w)*dem(c,w));
```

Display

trSD_L

trSD_R

trSD_N

trSD_P

****

Free Variable

Z;

Binary Variables

y(s)

x(d)

;

Positive Variable

u(s)

QSD(s,d)

QDC(d,c)

;

Equations

obj

cons1

cons2

cons3

```
cons4
cons5
;


scalar
say /1/;


scalar omega ;


*omega = card(S)*card(d)/2 ;


* 1 < gamma < aqrt(card(S)*card(d))


scalar teta ;
*teta=0;



Positive Variable
uu(s,d)
yy
;



equations
additional1
additional2;
```

```
obj..    z =l=  p*sum({d,c},QDC(d,c)) - (sum(d,f(d)*x(d)) + sum(s,A(s)*y(s)) +
sum({s,d},trSD_N(s,d)*QSD(s,d))

                                        + sum({d,c},trDC(d,c)*QDC(d,c)) + sum(s,b(s)*u(s))) -
(say*sum((s,d),trSD_P(s,d)*(QSD(s,d)-uu(s,d))) + omega*yy );

additional1(s,d)..       uu(s,d) =l= QSD(s,d);

additional2.. power(yy,2) =g= sum((s,d), power(trSD_P(s,d)*uu(s,d),2));

cons1(s)..       u(s) =L= capS(s)*y(s);

cons2(d)..       sum(S,QSD(s,d))=L= capD(d)*x(d);

cons3(s)..       u(s) =e= sum(d,QSD(s,d));

cons4(d)..       sum(s,QSD(s,d)) =e= sum(c,QDC(d,c));

cons5(c)..       sum(d,QDC(d,c)) =l= dem_N(c);



Model SCND
/
obj
cons1
```

```
cons2
cons3
cons4
cons5
additional1
additional2
/
;


Options
MIQCP = CPLEX
reslim =100
optcr = 0
;



set iter /1*11/;


parameter
tetaiter(iter)
Result(iter,*);


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

```
loop(iter,


teta=tetaiter(iter);
omega =  1+ teta*(sqrt(card(S)*card(d))-1) ;


Solve SCND us MIQCP max Z;


Result(iter,"Objective") = z.l;
Result(iter,"Teta")= Teta ;
Result(iter,"omega")= omega ;


Display
Z.l
y.l
x.l
QSD.l
QDC.l
yy.l
uu.l
omega
;
)
;


Display Result;
```

OR
OptimYar

SCND_BoxPolyhedral

```
Sets
S       /s1*s10/
D       /d1*d20/
C       /c1*c30/
w       /w1*w5/
;


Parameters
A(s)
f(d)
b(s)
trSD_L(s,d)
trSD_R(s,d)
trSD_N(s,d)
trSD_P(s,d)
trDC(d,c)
p
dem(c,w)
dem_N(c)
capD(d)
capS(s)
Prob(w)
/
w1      0.2
w2      0.3
w3      0.1
w4      0.2
```

```
w5      0.2
/


;



A(s)     = uniform(1000,1500);
f(d)     = uniform(2000,3000);
b(s)     = uniform(5,10);
trSD_L(s,d)= uniform(1,1.5);
trSD_R(s,d)= uniform(2,3);

trSD_N(s,d)= (trSD_L(s,d)+trSD_R(s,d))/2;
trSD_P(s,d)= (trSD_R(s,d)-trSD_L(s,d))/2;

trDC(d,c)= uniform(0.5,0.7);
p        = 15;
capD(d)  = uniform(500,1000);
capS(s)  = uniform(1000,2000);



dem(c,'w1')   = uniform(50,100);
dem(c,'w2')   = (1+0.8)*dem(c,'w1') ;
dem(c,'w3')   = (1+0.2)*dem(c,'w1') ;
dem(c,'w4')   = (1-0.2)*dem(c,'w1') ;
dem(c,'w5')   = (1-0.3)*dem(c,'w1') ;

dem_N(c)= sum(w,prob(w)*dem(c,w));
```

Display

trSD_L

trSD_R

trSD_N

trSD_P

****

Free Variable

Z;

Binary Variables

y(s)

x(d)

;

Positive Variable

u(s)

QSD(s,d)

QDC(d,c)

;

Equations

obj

cons1

cons2

cons3

cons4

cons5

;

scalar

say /1/;

scalar gamma ;

*gamma = card(S)*card(d)/2 ;

* 1 < gamma < ard(S)*card(d)

scalar teta ;

*teta=0;

Positive Variable

pp(s,d)

q

;

equations additional1;

```
obj..    z =l=  p*sum({d,c},QDC(d,c)) - (sum(d,f(d)*x(d)) + sum(s,A(s)*y(s)) +
sum({s,d},trSD_N(s,d)*QSD(s,d))

                                    + sum({d,c},trDC(d,c)*QDC(d,c)) + sum(s,b(s)*u(s))) -
(say*sum((s,d),pp(s,d))  + gamma*q );

additional1(s,d)..       pp(s,d)+q =g= trSD_P(s,d) *QSD(s,d);

cons1(s)..       u(s) =L= capS(s)*y(s);

cons2(d)..       sum(S,QSD(s,d))=L= capD(d)*x(d);

cons3(s)..       u(s) =e= sum(d,QSD(s,d));

cons4(d)..       sum(s,QSD(s,d)) =e= sum(c,QDC(d,c));

cons5(c)..       sum(d,QDC(d,c)) =l= dem_N(c);



Model SCND
/
obj
cons1
cons2
cons3
cons4
cons5
additional1
```

```
/
;

Options
mip = CPLEX
reslim =100
optcr = 0
;



set iter /1*11/;

parameter
tetaiter(iter)
Result(iter,*);

tetaiter(iter) = (ord(iter)-1)/(card(iter)-1);




loop(iter,

teta=tetaiter(iter);
gamma =  1+ teta*(card(S)*card(d)-1) ;

Solve SCND us mip max Z;
```

```
Result(iter,"Objective") = z.l;
Result(iter,"Teta")= Teta ;
Result(iter,"Gamma")= gamma ;


Display
Z.l
y.l
x.l
QSD.l
QDC.l
pp.l
q.l
gamma
;
)
;


Display Result;
```

OR
OptimYar

SCND_BoxPolyhedral_Case2

```
Sets
S       /s1*s10/
D       /d1*d20/
C       /c1*c30/
w       /w1*w5/
;


Parameters
A(s)
f(d)
b(s)
trSD_L(s,d)
trSD_R(s,d)
trSD_N(s,d)
trSD_P(s,d)
trDC(d,c)
p
dem(c,w)
dem_N(c)
capD(d)
capS(s)
Prob(w)
/
w1      0.2
w2      0.3
w3      0.1
w4      0.2
```

```
w5      0.2
/


;



A(s)    = uniform(1000,1500);
f(d)    = uniform(2000,3000);
b(s)    = uniform(5,10);
trSD_L(s,d)= uniform(1,1.5);
trSD_R(s,d)= uniform(2,3);

trSD_N(s,d)= (trSD_L(s,d)+trSD_R(s,d))/2;
trSD_P(s,d)= (trSD_R(s,d)-trSD_L(s,d))/2;

trDC(d,c)= uniform(0.5,0.7);
p       = 15;
capD(d)  = uniform(500,1000);
capS(s)  = uniform(1000,2000);



dem(c,'w1')   = uniform(50,100);
dem(c,'w2')   = (1+0.8)*dem(c,'w1') ;
dem(c,'w3')   = (1+0.2)*dem(c,'w1') ;
dem(c,'w4')   = (1-0.2)*dem(c,'w1') ;
dem(c,'w5')   = (1-0.3)*dem(c,'w1') ;

dem_N(c)= sum(w,prob(w)*dem(c,w));
```

Display

trSD_L

trSD_R

trSD_N

trSD_P

****

Free Variable

Z;

Binary Variables

y(s)

x(d)

;

Positive Variable

u(s)

QSD(s,d)

QDC(d,c)

;

Equations

obj

cons1

cons2

cons3

cons4

cons5

;

scalar

gamma /1/;

scalar say ;

* say > ard(S)*card(d) and say<1

scalar teta ;

*teta=0;

Positive Variable

pp(s,d)

q

;

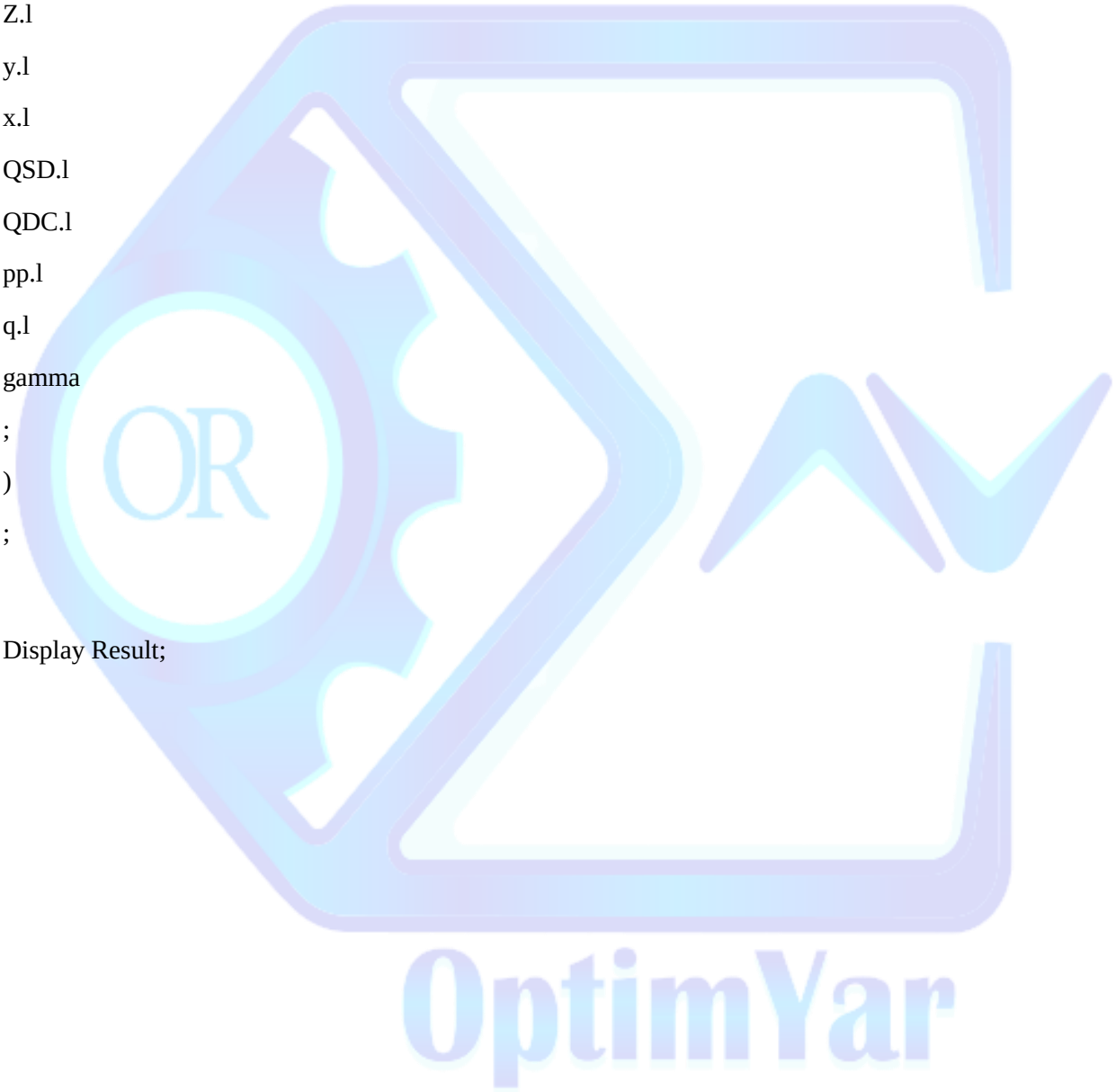
equations additional1;

obj.. z =l= p*sum({d,c},QDC(d,c)) - (sum(d,f(d)*x(d)) + sum(s,A(s)*y(s)) + sum({s,d},trSD_N(s,d)*QSD(s,d))

```
                                    + sum({d,c},trDC(d,c)*QDC(d,c)) + sum(s,b(s)*u(s))) -
(say*sum((s,d),pp(s,d))  + gamma*q );


additional1(s,d)..      pp(s,d)+q =g= trSD_P(s,d) *QSD(s,d);


cons1(s)..      u(s) =L= capS(s)*y(s);


cons2(d)..      sum(S,QSD(s,d))=L= capD(d)*x(d);


cons3(s)..      u(s) =e= sum(d,QSD(s,d));


cons4(d)..      sum(s,QSD(s,d)) =e= sum(c,QDC(d,c));


cons5(c)..      sum(d,QDC(d,c)) =l= dem_N(c);



Model SCND
/
obj
cons1
cons2
cons3
cons4
cons5
additional1
/
;
```

```
Options
mip = CPLEX
reslim =100
optcr = 0
;



set iter /1*11/;


parameter
tetaiter(iter)
Result(iter,*);

tetaiter(iter) = (ord(iter)-1)/(card(iter)-1);




loop(iter,

teta=tetaiter(iter);
say =  1/(card(S)*card(d)) + teta*(1-1/(card(S)*card(d))) ;

Solve SCND us mip max Z;

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

```
Result(iter,"Teta")= Teta ;
Result(iter,"say")= say ;


Display
Z.l
y.l
x.l
QSD.l
QDC.l
pp.l
q.l
gamma
;
)
;

Display Result;
```



www.optimyar.com

دوره جامع آنلاین بهینه‌سازی استوار و برنامه‌ریزی در شرایط عدم‌قطعیت 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 فرابتکاری، 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