Tiesinis programavimas

Pavyzdys. [3] 361 psl. Išspręskite TP bendrąjį uždavinį

Figure 1:

$$\begin{cases} 3x_1 - 4x_2 + x_3 = 2, \\ 2x_1 - 3x_2 + x_3 \le 3, \\ 4x_1 + 5x_2 + 3x_3 \ge 4, \\ x_1 + 4x_2 \le 2, \end{cases}$$

$$x_1 \le 0, \quad x_3 \ge 0,$$

$$7x_1 + 8x_2 - 9x_3 \to \min.$$

šiais būdais:
1. Pasinaudojant paketu "simplex" ir komanda "minimize_lp".
2. Uždavinį užrašant kanonine forma
   min c.x, kai A.x = b, x >= 0
   ir pasinaudojant komanda "linear_program".
3. Randant apribojimų srities kraštutinius taškus. Iš jų išrenkame tą, kuriame tikslo funkcijos reikšmė yra mažiausia.
4. Suvedant į dvimatį uždavinį, kurį sprendžiame grafiniu metodu. Su animacija parodome tikslo funkcijos lygio linijos judėjimą.
5. Pasinaudojant programų paketu "nopt".
6. Pasinaudojant programų paketu "COBYLA".
7. Sudarant ir išsprendžiant dualųjį uždavinį.

```
(%i1) f:7*x1+8*x2-9*x3;
```

```
(%i2) apr:[3*x1-4*x2+x3=2,2*x1-3*x2+x3<=3,4*x1+5*x2+3*x3>=4,x1+4*x2<=2,x1<=0,x3>=0];
```

(%o2) $[x3-4\,x2+3\,x1=2,\ x3-3\,x2+2\,x1<=3,\ 3\,x3+5\,x2+4\,x1>=4,\ 4\,x2+x1<=2,\ x1<=0,\ x3>=0]$

1. Sprendimas pasinaudojant paketu "simplex".

```
(%i3) load(simplex)$
```

```
(%i4) minimize_lp(f,apr);
```

(%o4) $[-\frac{111}{2},\ [x3=\frac{53}{12},\ x2=-\frac{7}{12},\ x1=-\frac{19}{12}]]$

http://www.personal.psu.edu/cxg286/Math484_V1.pdf

2. Uždavinį užrašome kanonine forma
min c.x, kai A.x = b, x >= 0
ir pasinaudojame komanda "linear_program":

Kadangi kintamasis x1<=0, tai jį pakeičiame x10=-x1>=0.
Kintamajam x2 nekeliamas joks apribojimas, todėl jį pakeičiame dviejų kintamųjų x20>=0 ir x200>=
Nelygybę 2*x1-3*x2+x3<=3 pakečiame lygybe 2*x1-3*x2+x3+x4=3, x4>=0.
Nelygybę 3*x3+5*x2+4*x1>=4 pakečiame lygybe 3*x3+5*x2+4*x1-x5=4, x5>=0.
Nelygybę 4*x2+x1<=2 pakečiame lygybe x1+4*x2+x6=2, x6>0.

```
(%i5) f2:subst([x1=-x10,x2=x20-x200],f),expand;
```

(%o5) $-9\,x3-8\,x200+8\,x20-7\,x10$

```
(%i6) apr2:subst([x1=-x10,x2=x20-x200],[apr[1],2*x1-3*x2+x3+x4=3,3*x3+5*x2+4*x1-x5=4,x1+4*x2+x6=
```

(%o6) $[x3+4\,x200-4\,x20-3\,x10=2,\ x4+x3+3\,x200-3\,x20-2\,x10=3,\ -x5+3\,x3-5\,x200+5\,x20-4\,x10=4,\ x6-4\,x200+4\,x20-x10=2]$

(%i7) `listofvars(apr2);`

(%o7) $[x10, x20, x200, x3, x4, x5, x6]$

Lygčių koeficientai prie šių kintamųjų sudaro matricą

(%i8) `A:matrix([-3,-4,4,1,0,0,0],[-2,-3,3,1,1,0,0],[-4,5,-5,3,0,-5,0],[-1,4,-4,0,0,0,1]);`

(%o8) $$\begin{bmatrix} -3 & -4 & 4 & 1 & 0 & 0 & 0 \\ -2 & -3 & 3 & 1 & 1 & 0 & 0 \\ -4 & 5 & -5 & 3 & 0 & -5 & 0 \\ -1 & 4 & -4 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Tikslo funkcijos koeficientai yra

(%i9) `c:[-7,8,-8,-9,0,0,0];`

(%o9) $[-7, 8, -8, -9, 0, 0, 0]$

Dešiniosios lygčių pusės yra

(%i10) `b:[2,3,4,2];`

(%o10) $[2, 3, 4, 2]$

Toliau sprendžiame su "linear_program":

(%i11) `spr:linear_program(A, b, c);`

(%o11) $[[\frac{19}{12}, 0, \frac{7}{12}, \frac{53}{12}, 0, 0, \frac{71}{12}], -\frac{111}{2}]$

(%i12) `s:spr[1];`

(%o12) $[\frac{19}{12}, 0, \frac{7}{12}, \frac{53}{12}, 0, 0, \frac{71}{12}]$

Grįžtame prie pradinių kintamųjų:

(%i13) `[x1=-s[1],x2=s[2]-s[3],x[3]=s[4]];`

(%o13) $[x1=-\frac{19}{12}, x2=-\frac{7}{12}, x_3=\frac{53}{12}]$

Atsakymas: f_min=-111/2, kai x1=-19/12, x2=-7/12, x[3]=53/12.

4. Pasinaudojant programų paketu "nopt"

(%i14) `f:7*x1+8*x2-9*x3;`

(%o14) $-9\,x3+8\,x2+7\,x1$

(%i15) `apr:[3*x1-4*x2+x3=2,2*x1-3*x2+x3<=3,4*x1+5*x2+3*x3>=4,x1+4*x2<=2,x1<=0,x3>=0];`

(%o15) $[x3-4\,x2+3\,x1=2, x3-3\,x2+2\,x1<=3, 3\,x3+5\,x2+4\,x1>=4, 4\,x2+x1<=2, x1<=0, x3>=0]$

(%i16) `load(nopt);`

(%o16) *C:/Users/aleksas/maxima/nopt.mac*

(%i17) `minimize_nopt(f,apr);`

(%o17) $[-\frac{111}{2}, [x1=-\frac{19}{12}, x2=-\frac{7}{12}, x3=\frac{53}{12}]]$

(%i18) `float(%);`

(%o18) $[-55.5, [x1=-1.583333333333333, x2=-0.5833333333333334, x3=4.416666666666667]]$

6. su COBYLA

(%i19) `f:7*x1+8*x2-9*x3;`

(%o19) $-9\,x3+8\,x2+7\,x1$

```
(%i20) apr:[3*x1-4*x2+x3=2,2*x1-3*x2+x3<=3,4*x1+5*x2+3*x3>=4,x1+4*x2<=2,x1<=0,x3>=0];
(%o20) [x3-4 x2+3 x1=2,x3-3 x2+2 x1<=3,3 x3+5 x2+4 x1>=4,4 x2+x1<=2,x1<=0,x3>=0]
```

```
(%i38) load(fmin_cobyla)$
```

```
(%i22) fmin_cobyla(f,[x1,x2,x3],[0,0,0], constraints = apr, iprint=1);
   Normal return from subroutine COBYLA
   NFVALS =    37    F =-5.550000E+01    MAXCV = 1.776357E-15
   X =-1.583333E+00  -5.833333E-01   4.416667E+00
(%o22) [[x1=-1.583333333333332,x2=-0.5833333333333328,x3=4.416666666666663],-
55.49999999999996,37,0]
```

5. Sprendimas, randant kraštutinius taškus:

Apibrėžiame funkciją "ext", kuri randa srities kraštutinius taškus(extreme points).
Tikslo funkcijos minimumas pasiekiamas šiuose taškuose.
Toliau toliau iš jų atrenkame tuos, kuriuose pasiekiamas minimumas.

```
(%i23) kill(all)$
```

```
(%i1) load(simplex)$
```

```
(%i2) ext(apr):=block([var,fs,cs,ap,s,S,m],
      var:sort(listofvars(apr)),
      s:apply("+",var),
      fs:append([1,s,-s],var,-var),
      ap(k):=subst(apr[k]=(lhs(apr[k])=rhs(apr[k])),apr),
      cs:makelist(ap(k),k,1,length(apr)),
      S:[],
      for f in fs do
      for c in cs do
      (
      m:minimize_lp(f,c),
      if listp(m) then
      S:cons(subst(m[2],var),S)
      ),
      listify(setify(S))
      )$
```

```
(%i3) f:7*x1+8*x2-9*x3;
(%o3) -9 x3+8 x2+7 x1
```

```
(%i4) apr:[3*x1-4*x2+x3=2,2*x1-3*x2+x3<=3,4*x1+5*x2+3*x3>=4,x1+4*x2<=2,x1<=0,x3>=0]$
```

```
(%i5) T:ext(apr);
```

(%o5) $\left[\left[-\frac{19}{12},-\frac{7}{12},\frac{53}{12}\right],\left[-\frac{2}{5},\frac{3}{5},\frac{28}{5}\right],\left[0,-\frac{2}{17},\frac{26}{17}\right],\left[0,\frac{1}{2},4\right]\right]$

Apskaičiuosime tikslo funkcijos reikšmes šiuose taškuose.

```
(%i6) r:makelist(ev(f,[x1=T[k][1],x2=T[k][2],x3=T[k][3]]),k,1,length(T));
```

(%o6) $\left[-\frac{111}{2},-\frac{242}{5},-\frac{250}{17},-32\right]$

Mažiausioji reikšmė yra:

```
(%i7) m:lmin(%);
```

(%o7) $-\frac{111}{2}$

Ji įgyjama taškuose:

```
(%i8) sublist_indices (r,lambda([x],x=m))$
      min_taskai:makelist(T[k],k,%);
```

(%o9) $[[-\frac{19}{12},-\frac{7}{12},\frac{53}{12}]]$

Atsakymas: f_min=-111/2, kai [x1,x2,x3]=[-19/12,-7/12,53/12].

6. Suvedant į dvimatį uždavinį, kurį sprendžiame grafiniu metodu.
   Su animacija parodome tikslo funkcijos lygio linijos judėjimą.

```
(%i10) f:7*x1+8*x2-9*x3;
```

(%o10) $-9\ x3+8\ x2+7\ x1$

```
(%i11) apr:[3*x1-4*x2+x3=2,2*x1-3*x2+x3<=3,4*x1+5*x2+3*x3>=4,x1+4*x2<=2,x1<=0,x3>=0]$
```

Eliminuosime kintamąjį x3.

```
(%i12) s3:solve(apr[1],x3);
```

(%o12) $[x3=4\ x2-3\ x1+2]$

```
(%i13) subst(%,apr);
```

(%o13) $[2=2,x2-x1+2<=3,3\left(4\ x2-3\ x1+2\right)+5\ x2+4\ x1>=4,4\ x2+x1<=2,x1<=0,4\ x2-3\ x1+2>=0]$

```
(%i14) apr1:expand(%);
```

(%o14) $[2=2,x2-x1+2<=3,17\ x2-5\ x1+6>=4,4\ x2+x1<=2,x1<=0,4\ x2-3\ x1+2>=0]$

```
(%i15) f1:subst(s3,f),expand;
```

(%o15) $-28\ x2+34\ x1-18$

```
(%i36) load(draw)$
       ratprint:false$
```

```
(%i18) set_draw_defaults(
                 x_voxel = 40,
                 y_voxel = 40,
                 xrange = [-2,1],
                 yrange = [-1,1],
                 grid   = true,
                 proportional_axes = xy,
                 fill_color = skyblue)$
```

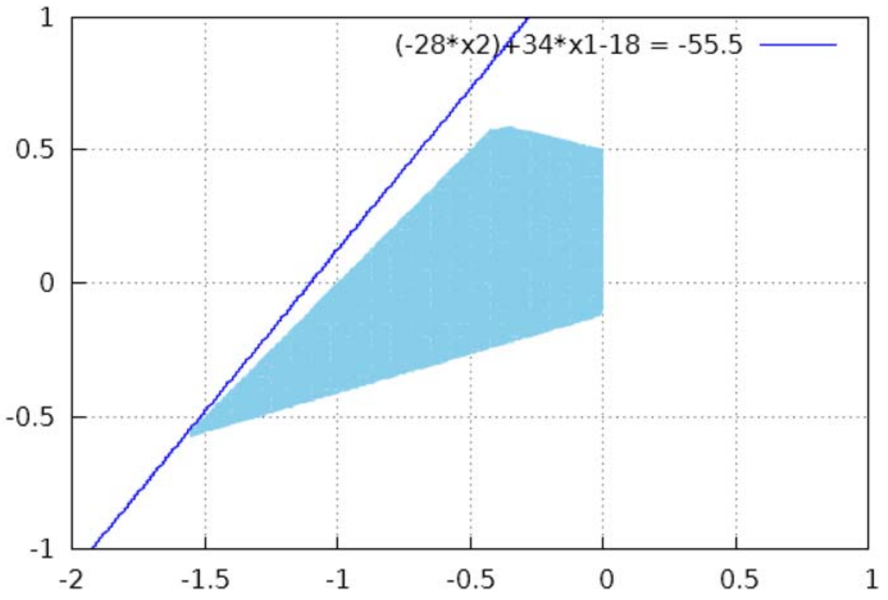
```
(%i19) sritis:apply("and", apr1);
```

(%o19) $x2-x1+2<=3\ \mathit{and}\ 17\ x2-5\ x1+6>=4\ \mathit{and}\ 4\ x2+x1<=2\ \mathit{and}\ x1<=0\ \mathit{and}\ 4\ x2-3\ x1+2>=0$

Brėžinys su animacija:

(%i20)
```
with_slider_draw(
        z, makelist(i, i, -55.5, 0,10),
        region(sritis, x1, -2, 1, x2, -1, 1),
        key   = string(ev(f1,nouns)=z),
        implicit(f1=z,x1,-2,1,x2, -1, 1));
```

(%t20)



(%o20)

Matome, kad minimumas yra pasiekiamas tiesių x2-x1+2=3, 17*x2-5*x1+6=4 susikirtimo taške.

(%i21)
```
sol:solve([x2-x1+2=3,17*x2-5*x1+6=4]);
```

(%o21) $[[x2=-\frac{7}{12}, x1=-\frac{19}{12}]]$

(%i22)
```
float(%[1]);
```

(%o22) $[x2=-0.5833333333333334, x1=-1.583333333333333]$

(%i23)
```
subst(sol[1],s3);
```

(%o23) $[x3=\frac{53}{12}]$

(%i24)
```
float(%[1]);
```

(%o24) $x3=4.416666666666667$

7. Sudarant ir išsprendžiant dualųjį uždavinį.
Bendrojo TP uždavinio dualaus uždavinio formulavimą žr. [2], 83 psl.; [5]

(%i25)
```
dual(A,b,c,I0,J0):=block([m,n,I,J,JminusJ0,Y],
m:matrix_size(A)[1],
n:matrix_size(A)[2],
I:makelist(k,k,1,m),
J:makelist(k,k,1,n),
JminusJ0:sublist(J,lambda([x],not member(x,J0))),
Y:makelist(concat(y,k),k,I),
[b.Y,append(
makelist((transpose(A).Y)[j,1]<=c[j],j,J0),
makelist((transpose(A).Y)[j,1]=c[j],j,JminusJ0),
makelist(Y[i]>=0,i,I0))]
)$
```

(%i26)
```
I0:[2,3,4,5];
```

(%o26) [2,3,4,5]

(%i27)
```
J0:[3];
```

(%o27) [3]

```
(%i28) b:[2,-3,4,-2,0];
(%o28) [2,-3,4,-2,0]
```

```
(%i29) c:[7,8,-9];
(%o29) [7,8,-9]
```

```
(%i30) A:matrix([3,-4,1],[-2,3,-1],[4,5,3],[-1,-4,0],[-1,0,0]);
```

(%o30) $\begin{bmatrix} 3 & -4 & 1 \\ -2 & 3 & -1 \\ 4 & 5 & 3 \\ -1 & -4 & 0 \\ -1 & 0 & 0 \end{bmatrix}$

Dualusis uždavinys:

```
(%i31) dp:dual(A,b,c,I0,J0);
```

(%o31) [$-2\,y4+4\,y3-3\,y2+2\,y1$, [$3\,y3-y2+y1$<=-9, $-y5-y4+4\,y3-2\,y2+3\,y1=7$, $-4\,y4+5\,y3+3\,y2-4\,y1=8$, $y2$>=0, $y3$>=0, $y4$>=0, $y5$>=0]]

```
(%i32) f1:dp[1];
```

(%o32) $-2\,y4+4\,y3-3\,y2+2\,y1$

```
(%i33) apr1:dp[2];
```

(%o33) [$3\,y3-y2+y1$<=-9, $-y5-y4+4\,y3-2\,y2+3\,y1=7$, $-4\,y4+5\,y3+3\,y2-4\,y1=8$, $y2$>=0, $y3$>=0, $y4$>=0, $y5$>=0]

```
(%i34) load(simplex);
(%o34)
C:/Program Files (x86)/Maxima-sbcl-5.36.1/share/maxima/5.36.1/share/simplex/simplex.mac
```

```
(%i35) maximize_lp(f1,apr1);
```

(%o35) $[-\frac{111}{2}, [y5=0, y4=0, y3=\frac{1}{2}, y2=\frac{73}{2}, y1=26]]$

[1] A. Apynis, E. Stankus, Matematikos pagrindai, V., TEV, 2009
[2] A. Apynis, Optimizavimo metodai, V., VU, 2005
[3] V. Pekarskas, A.Pekarskienė, Tiesinės algebros ir analizinės geometrijos elementai, 2004
[4] V. Čiočys, R.Jasiulionis, Matematinis programavimas, V., Mokslas, 1990
[5] https://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/15859-f11/www/notes/lecture05.pdf