

## Metoda st elby

St elba pro soustavu rovnic

$$y1'=f(x,y2,y2)$$

$$y2'=g(x,y2,y2)$$

a okrajové podmínky  $y1(a) + y2(a) = 1, y1(b) + y2(b) = 2$

## St elba (diferen ní náhrada)

```
Strelba1 := proc (f, g, a, b, alfa1, alfa2, beta1, beta2, gama1, gama2, eps, h, z0, Lx)
  local z, zn, s, i, res, res1, res2, y1, y2, x, n, g1, g2, v1, v2;
  n := numelems(Lx);
  z := z0;
  s := 999999999;
  i := 0;
  printf("iterace   zn   sn\n");
  printf("-----\n");
  printf("%5d %14.9f \n", i, evalf(z));
  while eps < s and i < 10 do
    if abs(beta1) < 0.00000001 then
      res1 := dsolve({diff(y1(x), x) = f(x, y1(x), y2(x)), diff(y2(x), x) = g(x, y1(x),
y2(x)), y1(a) = (gama1-beta1*z)/alfa1, y2(a) = z}, {y1(x), y2(x)}, type = numeric);
      res2 := dsolve({diff(y1(x), x) = f(x, y1(x), y2(x)), diff(y2(x), x) = g(x, y1(x),
y2(x)), y1(a) = (gama1-beta1*(z+h))/alfa1, y2(a) = z+h}, {y1(x), y2(x)}, type
= numeric);
      else
      res1 := dsolve({diff(y1(x), x) = f(x, y1(x), y2(x)), diff(y2(x), x) = g(x, y1(x),
y2(x)), y1(a) = z, y2(a) = (gama1-alfa1*z)/beta1}, {y1(x), y2(x)}, type = numeric);
      res2 := dsolve({diff(y1(x), x) = f(x, y1(x), y2(x)), diff(y2(x), x) = g(x, y1(x),
y2(x)), y1(a) = z+h, y2(a) = (gama1-alfa1*(z+h))/beta1}, {y1(x), y2(x)}, type
= numeric);
      end if;
      zn := z - (alfa2*subs(res1(b), y1(x)) + beta2*subs(res1(b), y2(x)) - gama2)*h
/ ((alfa2*subs(res2(b), y1(x)) + beta2*subs(res2(b), y2(x))) - (alfa2*subs(res1(b),
y1(x)) + beta2*subs(res1(b), y2(x))));
      s := abs(zn-z);
      z := zn;
      i := i + 1;
      printf("%5d %14.9f %14.9f\n", i, evalf(z), evalf(s));
    end do;
    if abs(beta1) < 0.00000001 then
      res := dsolve({diff(y1(x), x) = f(x, y1(x), y2(x)), diff(y2(x), x) = g(x, y1(x),
y2(x)), y1(a) = (gama1-beta1*z)/alfa1, y2(a) = z}, {y1(x), y2(x)}, type = numeric);
    else
      res := dsolve({diff(y1(x), x) = f(x, y1(x), y2(x)), diff(y2(x), x) = g(x, y1(x),
y2(x)), y1(a) = z, y2(a) = (gama1-alfa1*z)/beta1}, {y1(x), y2(x)}, type = numeric);
    end if;
    g1 := plots[odeplot](res, [[x, y1(x)], 0..1]);
    g2 := plots[odeplot](res, [[x, y2(x)], 0..1]);
    v1 := [seq([Lx[i], subs(res1(Lx[i]), y1(x))], i = 1..n)]; print[v1];
    v2 := [seq([Lx[i], subs(res1(Lx[i]), y2(x))], i = 1..n)];
    RETURN(g1, g2, v1, v2)
  end proc;
```

## [ St elba (varia ní rovnice)

```

Strelba2 := proc (f, g, a, b, alfa1, alfa2, beta1, beta2, gama1, gama2, eps, z0, Lx)
  local z, zn, s, i, res, res1, res2, y1, y2, p1, p2, x, n, g1, g2, v1, v2, f1, f2, gg1, gg2, rov1, rov2,
  rov3, rov4;
  n := numelems(Lx);
  z := z0;
  s := 999999999;
  i := 0;
  printf("iterace    zn    sn\n");
  printf("-----\n");
  printf("%5d %14.9f \n", i, evalf(z));
  f1 := D[2](f);
  f2 := D[3](f);
  g1 := D[2](g);
  g2 := D[3](g);
  rov1 := diff(y1(x), x) = f(x, y1(x), y2(x));
  rov2 := diff(y2(x), x) = g(x, y1(x), y2(x));
  rov3 := diff(p1(x), x) = f1(x, y1(x), y2(x)) * p1(x) + f2(x, y1(x), y2(x)) * p2(x);
  rov4 := diff(p2(x), x) = g1(x, y1(x), y2(x)) * p1(x) + g2(x, y1(x), y2(x)) * p2(x);
  while eps < s and i < 10 do
    if abs(beta1) < 0.00000001 then
      res1 := dsolve( {rov1, rov2, rov3, rov4, y1(a) = (gama1 - beta1 * z) / alfa1, y2(a) = z,
      p1(a) = -beta1 / alfa1, p2(a) = 1.0}, {y1(x), y2(x), p1(x), p2(x)}, type = numeric);
    else
      res1 := dsolve( {rov1, rov2, rov3, rov4, y1(a) = z, y2(a) = (gama1 - alfa1 * z) / beta1,
      p1(a) = 1.0, p2(a) = -alfa1 / beta1}, {y1(x), y2(x), p1(x), p2(x)}, type = numeric);
    end if;
    zn := z - (alfa2 * subs(res1(b), y1(x)) + beta2 * subs(res1(b), y2(x)) - gama2) / ((alfa2
    * subs(res1(b), p1(x)) + beta2 * subs(res1(b), p2(x))));
    s := abs(zn - z);
    z := zn;
    i := i + 1;
    printf("%5d %14.9f %14.9f\n", i, evalf(z), evalf(s));
  end do;
  if abs(beta1) < 0.00000001 then
    res := dsolve( {rov1, rov2, y1(a) = (gama1 - beta1 * z) / alfa1, y2(a) = z }, {y1(x),
    y2(x)}, type = numeric);
  else
    res := dsolve( {rov1, rov2, y1(a) = z, y2(a) = (gama1 - alfa1 * z) / beta1}, {y1(x),
    y2(x)}, type = numeric);
  end if;
  gg1 := plots[odeplot](res, [[x, y1(x)], 0 .. 1]);
  gg2 := plots[odeplot](res, [[x, y2(x)], 0 .. 1]);
  v1 := [seq([Lx[i], subs(res1(Lx[i]), y1(x))], i = 1 .. n)]; print[v1];
  v2 := [seq([Lx[i], subs(res1(Lx[i]), y2(x))], i = 1 .. n)];
  RETURN(gg1, gg2, v1, v2)
end proc:

```



# Diferen ní náhrada derivace podle parametru

## ▼ P íklad 1

```
> f := (x, y1, y2) → y2;  
g := (x, y1, y2) → y1 · y1;  
f := (x, y1, y2) → y2  
g := (x, y1, y2) → y1 y1
```

(1.1)

```
> a := 0;  
b := 1;  
alfa1 := 1;  
alfa2 := 1;  
beta1 := 0;  
beta2 := 0;  
gama1 := 1;  
gama2 := 1;  
eps := 0.000001;  
h := 0.001;  
z0 := 0.0;  
m := 10;  
Lx := evalf([seq( $\frac{i}{m}$ , i = 1 .. m)]);  
a := 0  
b := 1  
alfa1 := 1  
alfa2 := 1  
 $\beta$ 1 := 0  
 $\beta$ 2 := 0  
gama1 := 1  
gama2 := 1  
eps := 0.000001  
h := 0.001  
z0 := 0.  
m := 10
```

```
Lx := [0.1000000000, 0.2000000000, 0.3000000000, 0.4000000000, 0.5000000000,  
0.6000000000, 0.7000000000, 0.8000000000, 0.9000000000, 1.]
```

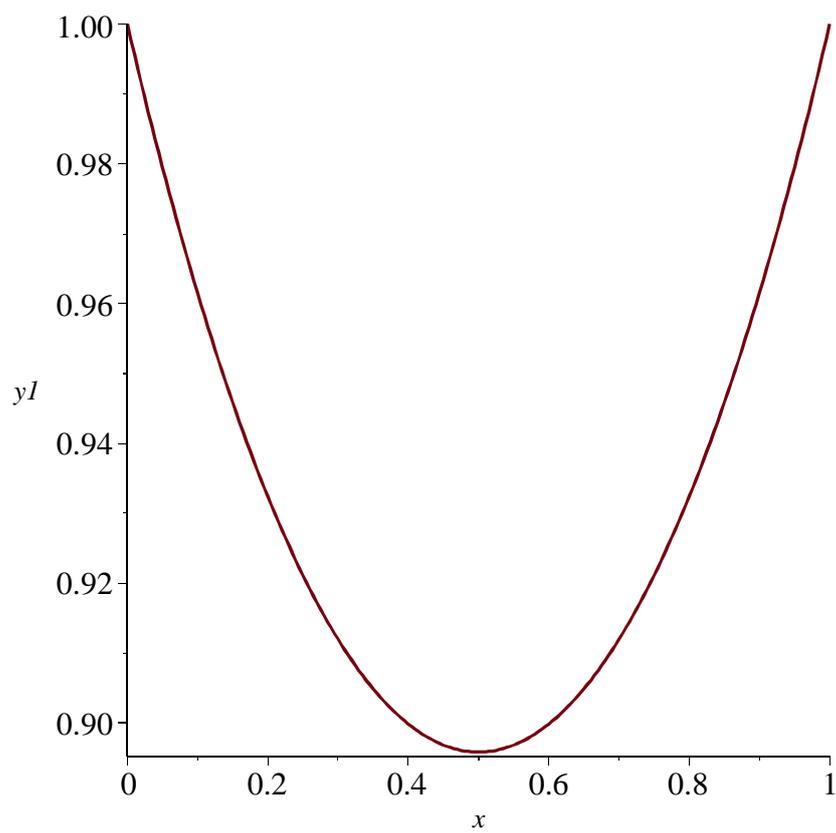
(1.2)

```
> v := Strelba1(f, g, a, b, alfa1, alfa2, beta1, beta2, gama1, gama2, eps, h, z0, Lx) :  
iterace                    zn                    sn
```

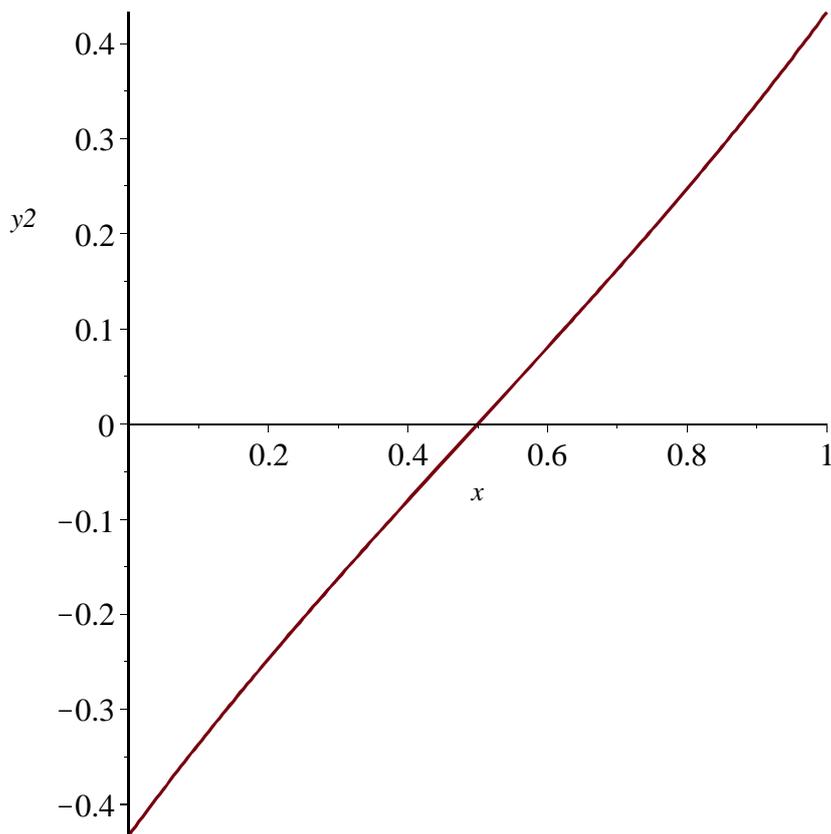
```
-----  
0            0.000000000  
1           -0.417514345           0.417514345  
2           -0.432914137           0.015399791  
3           -0.432935489           0.000021353  
4           -0.432935491           0.000000002
```

```
> # Graf funkce y1(x)
```

```
> v[1];
```



```
> # Graf funkce y2(x)  
> v[2];
```



```
> # Tabulka hodnot funkce y1(x)
```

```
> linalg[matrix](v[3]);
```

```
0.1000000000 0.961571688443231  
0.2000000000 0.932406313536387  
0.3000000000 0.911949313555517  
0.4000000000 0.899821932344128  
0.5000000000 0.895803615162645  
0.6000000000 0.899821938108504  
0.7000000000 0.911949325849244  
0.8000000000 0.932406328415301  
0.9000000000 0.961571698284689  
1.          1.00000000245791
```

```
> # Tabulka hodnot funkce y2(x)
```

```
> linalg[matrix](v[4]);
```

(1.3)

0.1000000000	-0.336885924732912
0.2000000000	-0.247341228307212
0.3000000000	-0.162426967184098
0.4000000000	-0.0804865019588489
0.5000000000	5.40986695219715 10 <sup>-8</sup>
0.6000000000	0.0804866199310347
0.7000000000	0.162427104904757
0.8000000000	0.247341370303990
0.9000000000	0.336886053392418
1.	0.432935571555822

(1.4)

Náhrada derivace podle parametru pomocí variačních rovnic

**Příklad 1**

```
> f := (x, y1, y2) -> y2;
   g := (x, y1, y2) -> y1 * y1;
```

```
f := (x, y1, y2) -> y2
g := (x, y1, y2) -> y1 * y1
```

(2.1)

```
> a := 0;
   b := 1;
   alfa1 := 1;
   alfa2 := 1;
   beta1 := 0;
   beta2 := 0;
   gama1 := 1;
   gama2 := 1;
   eps := 0.000001;
   z0 := 0.0;
   m := 10;
   Lx := evalf([seq(i/m, i = 1..m)]);
```

```
a := 0
b := 1
alfa1 := 1
alfa2 := 1
beta1 := 0
beta2 := 0
gama1 := 1
gama2 := 1
eps := 0.000001
z0 := 0.
m := 10
```

```
Lx := [0.1000000000, 0.2000000000, 0.3000000000, 0.4000000000, 0.5000000000,
```

(2.2)

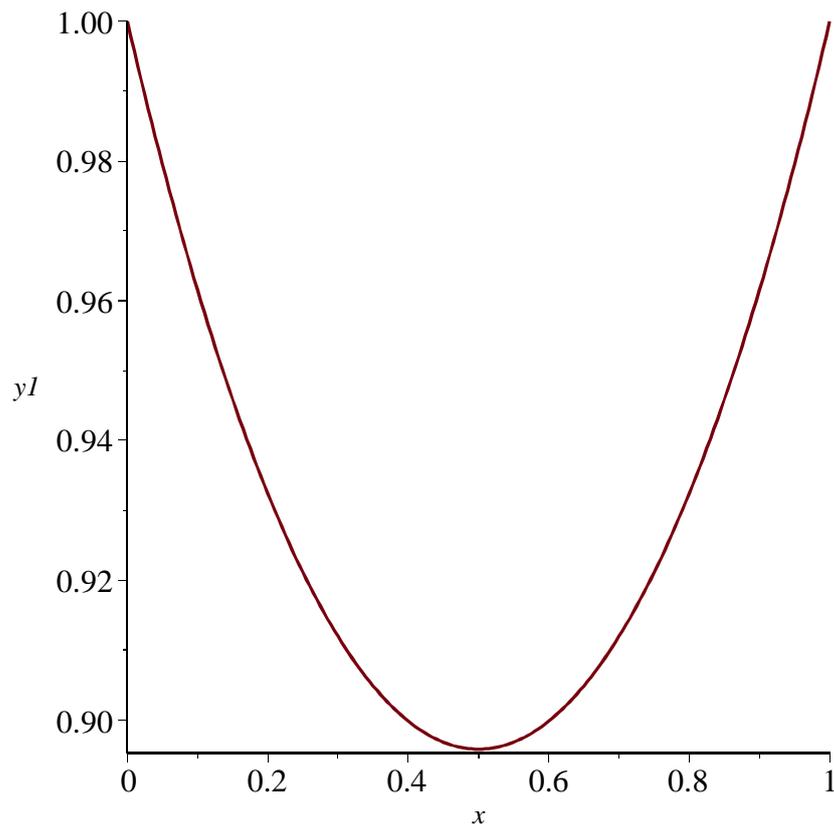
```
0.6000000000, 0.7000000000, 0.8000000000, 0.9000000000, 1.]
```

```
> v := Strelba2(f, g, a, b, alfa1, alfa2, beta1, beta2, gama1, gama2, eps, z0, Lx) :
```

```
iterace      zn      sn
-----
0      0.000000000
1     -0.417549745    0.417549745
2     -0.432915532    0.015365788
3     -0.432935497    0.000019964
4     -0.432935497    0.000000000
```

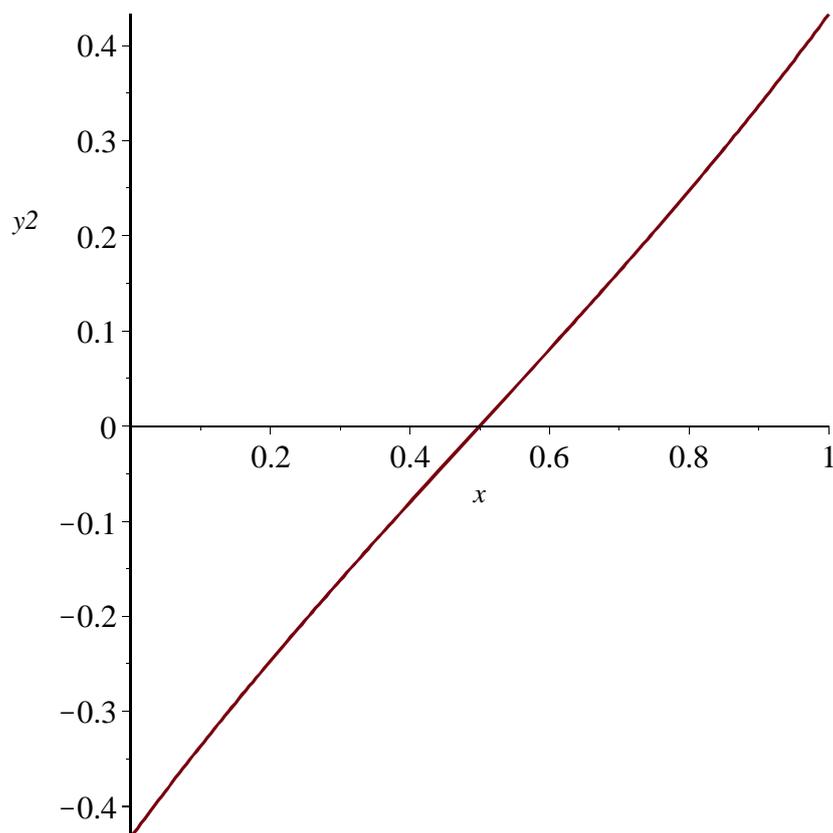
```
> # Graf funkce y1(x)
```

```
> v[1];
```



```
> # Graf funkce y2(x)
```

```
> v[2];
```



```
> # Tabulka hodnot funkce y1(x)
```

```
> linalg[matrix](v[3]);
```

0.1000000000	0.961571690041638
0.2000000000	0.932406320053545
0.3000000000	0.911949320211793
0.4000000000	0.899821935042116
0.5000000000	0.895803615033868
0.6000000000	0.899821936207792
0.7000000000	0.911949321560755
0.8000000000	0.932406320451273
0.9000000000	0.961571687412824
1.	1.00000000004551

(2.3)

```
> # Tabulka hodnot funkce y2(x)
```

```
> linalg[matrix](v[4]);
```

|

0.1000000000	-0.336885977162505
0.2000000000	-0.247341281966862
0.3000000000	-0.162427015389904
0.4000000000	-0.0804865329115788
0.5000000000	$3.51238581015184 \cdot 10^{-8}$
0.6000000000	0.0804866013500601
0.7000000000	0.162427083062756
0.8000000000	0.247341356957932
0.9000000000	0.336886060295347
1.	0.432935613823946

**(2.4)**