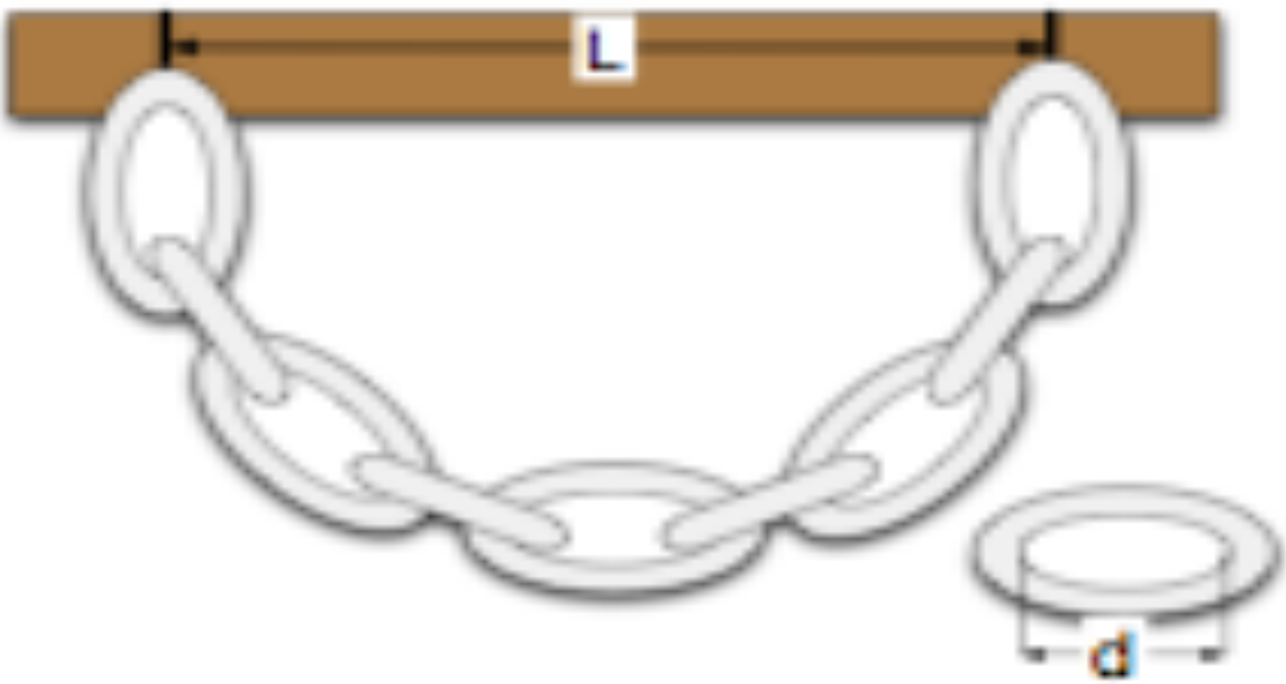
Example of Constrained Minimazion

(catenary)

```
> restart:
```




```
> with(plots) :
  with(LinearAlgebra):
> N := 10 ;
  L := 0.1 ;
  d := 0.015 ;
```

$$N := 10$$
$$L := 0.1$$
$$d := 0.015 \quad (1)$$

The potential enery

```
> f := sum( y[k], k=1..N-1 ) ;
```

$$f := y_1 + y_2 + y_3 + y_4 + y_5 + y_6 + y_7 + y_8 + y_9 \quad (2)$$

The constraints

```
> h1 := y[0] ;
  h2 := y[N] ;
```

$$h1 := y_0$$
$$h2 := y_{10} \quad (3)$$

```
> h3 := x[0] ;
  h4 := x[N] - L ;
```

$$h3 := x_0$$

(4)

$$h4 := x_{10} - 0.1 \quad (4)$$

```
> for k from 1 to N do
    h||(k+4)  := (x[k]-x[k-1])^2 + (y[k]-y[k-1])^2 - d^2 ;
  end;
```

$$h5 := (x_1 - x_0)^2 + (y_1 - y_0)^2 - 0.000225$$
$$h6 := (x_2 - x_1)^2 + (y_2 - y_1)^2 - 0.000225$$
$$h7 := (x_3 - x_2)^2 + (y_3 - y_2)^2 - 0.000225$$
$$h8 := (x_4 - x_3)^2 + (y_4 - y_3)^2 - 0.000225$$
$$h9 := (x_5 - x_4)^2 + (y_5 - y_4)^2 - 0.000225$$
$$h10 := (x_6 - x_5)^2 + (y_6 - y_5)^2 - 0.000225$$
$$h11 := (x_7 - x_6)^2 + (y_7 - y_6)^2 - 0.000225$$
$$h12 := (x_8 - x_7)^2 + (y_8 - y_7)^2 - 0.000225$$
$$h13 := (x_9 - x_8)^2 + (y_9 - y_8)^2 - 0.000225$$
$$h14 := (x_{10} - x_9)^2 + (y_{10} - y_9)^2 - 0.000225 \quad (5)$$

```
> #with(Optimization):
> #SOL := Minimize( f, {seq(h||k=0,k=1..14)} ) ;
> #plot( subs(SOL[2],[seq([x[k],y[k]],k=0..N)]), scaling=CONSTRAINED)
  ;
```

Build the nonlinear system using Lagrange Multiplier

```
> Lag := f-add(lambda[k]*h||k,k=1..N+4) ;
```

$$Lag := y_1 + y_2 + y_3 + y_4 + y_5 + y_6 + y_7 + y_8 + y_9 - \lambda_1 y_0 - \lambda_2 y_{10} - \lambda_3 x_0 - \lambda_4 (x_{10} - 0.1) - \lambda_5 \left((x_1 - x_0)^2 + (y_1 - y_0)^2 - 0.000225 \right) - \lambda_6 \left((x_2 - x_1)^2 + (y_2 - y_1)^2 - 0.000225 \right) - \lambda_7 \left((x_3 - x_2)^2 + (y_3 - y_2)^2 - 0.000225 \right) - \lambda_8 \left((x_4 - x_3)^2 + (y_4 - y_3)^2 - 0.000225 \right) - \lambda_9 \left((x_5 - x_4)^2 + (y_5 - y_4)^2 - 0.000225 \right) - \lambda_{10} \left((x_6 - x_5)^2 + (y_6 - y_5)^2 - 0.000225 \right) - \lambda_{11} \left((x_7 - x_6)^2 + (y_7 - y_6)^2 - 0.000225 \right) - \lambda_{12} \left((x_8 - x_7)^2 + (y_8 - y_7)^2 - 0.000225 \right) - \lambda_{13} \left((x_9 - x_8)^2 + (y_9 - y_8)^2 - 0.000225 \right) - \lambda_{14} \left((x_{10} - x_9)^2 + (y_{10} - y_9)^2 - 0.000225 \right) \quad (6)$$

```
> EQS := [seq(diff(Lag,x[k]),k=0..N),
          seq(diff(Lag,y[k]),k=0..N),
          seq(diff(Lag,lambda[k]),k=1..N+4)];
```

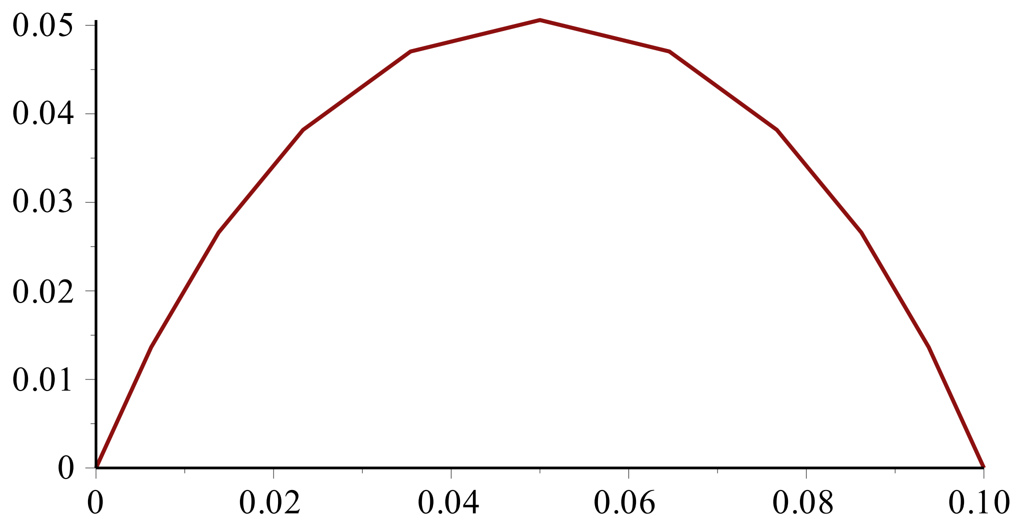
$$EQS := \left[-\lambda_3 - \lambda_5 (-2 x_1 + 2 x_0), -\lambda_5 (2 x_1 - 2 x_0) - \lambda_6 (-2 x_2 + 2 x_1), -\lambda_6 (2 x_2 - 2 x_1) - \lambda_7 (-2 x_3 + 2 x_2), -\lambda_7 (2 x_3 - 2 x_2) - \lambda_8 (-2 x_4 + 2 x_3), -\lambda_8 (2 x_4 - 2 x_3) - \lambda_9 (-2 x_5 + 2 x_4), -\lambda_9 (2 x_5 - 2 x_4) - \lambda_{10} (-2 x_6 + 2 x_5), -\lambda_{10} (2 x_6 \right. \quad (7)$$

$$- 2x_5) - \lambda_{11}(-2x_7 + 2x_6), -\lambda_{11}(2x_7 - 2x_6) - \lambda_{12}(-2x_8 + 2x_7), -\lambda_{12}(2x_8 - 2x_7) - \lambda_{13}(-2x_9 + 2x_8), -\lambda_{13}(2x_9 - 2x_8) - \lambda_{14}(-2x_{10} + 2x_9), -\lambda_4 - \lambda_{14}(2x_{10} - 2x_9), -\lambda_1 - \lambda_5(-2y_1 + 2y_0), 1 - \lambda_5(2y_1 - 2y_0) - \lambda_6(-2y_2 + 2y_1), 1 - \lambda_6(2y_2 - 2y_1) - \lambda_7(-2y_3 + 2y_2), 1 - \lambda_7(2y_3 - 2y_2) - \lambda_8(-2y_4 + 2y_3), 1 - \lambda_8(2y_4 - 2y_3) - \lambda_9(-2y_5 + 2y_4), 1 - \lambda_9(2y_5 - 2y_4) - \lambda_{10}(-2y_6 + 2y_5), 1 - \lambda_{10}(2y_6 - 2y_5) - \lambda_{11}(-2y_7 + 2y_6), 1 - \lambda_{11}(2y_7 - 2y_6) - \lambda_{12}(-2y_8 + 2y_7), 1 - \lambda_{12}(2y_8 - 2y_7) - \lambda_{13}(-2y_9 + 2y_8), 1 - \lambda_{13}(2y_9 - 2y_8) - \lambda_{14}(-2y_{10} + 2y_9), -\lambda_2 - \lambda_{14}(2y_{10} - 2y_9), -y_0, -y_{10}, -x_0, -x_{10} + 0.1, -(x_1 - x_0)^2 - (y_1 - y_0)^2 + 0.000225, -(x_2 - x_1)^2 - (y_2 - y_1)^2 + 0.000225, -(x_3 - x_2)^2 - (y_3 - y_2)^2 + 0.000225, -(x_4 - x_3)^2 - (y_4 - y_3)^2 + 0.000225, -(x_5 - x_4)^2 - (y_5 - y_4)^2 + 0.000225, -(x_6 - x_5)^2 - (y_6 - y_5)^2 + 0.000225, -(x_7 - x_6)^2 - (y_7 - y_6)^2 + 0.000225, -(x_8 - x_7)^2 - (y_8 - y_7)^2 + 0.000225, -(x_9 - x_8)^2 - (y_9 - y_8)^2 + 0.000225, -(x_{10} - x_9)^2 - (y_{10} - y_9)^2 + 0.000225]$$

```
> SOL := fsolve( EQS, {seq(x[k]=L*k/N,k=0..N),seq(y[k]=k*(N-k),k=0..
  N),seq(lambda[k]=0,k=1..N+4)} );
```

$$SOL := \{\lambda_1 = 4.500000000, \lambda_2 = 4.500000000, \lambda_3 = 2.051051014, \lambda_4 = -2.051051014, \lambda_5 = 164.8460907, \lambda_6 = 135.2233143, \lambda_7 = 107.7899720, \lambda_8 = 84.70084785, \lambda_9 = 70.37052937, \lambda_{10} = 70.37052937, \lambda_{11} = 84.70084785, \lambda_{12} = 107.7899720, \lambda_{13} = 135.2233143, \lambda_{14} = 164.8460907, x_0 = 0., x_1 = 0.006221109052, x_2 = 0.01380504909, x_3 = 0.02331915775, x_4 = 0.03542677572, x_5 = 0.05000000000, x_6 = 0.06457322428, x_7 = 0.07668084225, x_8 = 0.08619495091, x_9 = 0.09377889095, x_{10} = 0.1000000000, y_0 = 0., y_1 = 0.01364909529, y_2 = 0.02659065059, y_3 = 0.03818727666, y_4 = 0.04704196961, y_5 = 0.05059459316, y_6 = 0.04704196961, y_7 = 0.03818727666, y_8 = 0.02659065059, y_9 = 0.01364909529, y_{10} = 0.\} \quad \textbf{(8)}$$

```
> plot( subs(SOL,[seq([x[k],y[k]],k=0..N)]), scaling=CONSTRAINED);
```



Check if this solution is a max or a min

Compute the hessian of Lag at the point

```
> X      := [seq(x[k],k=0..N),seq(y[k],k=0..N)] ;
  LAMBDA := [seq(lambda[k],k=1..N+4)] ;
```

$$X := [x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, y_0, y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8, y_9, y_{10}]$$

$$LAMBDA := [\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6, \lambda_7, \lambda_8, \lambda_9, \lambda_{10}, \lambda_{11}, \lambda_{12}, \lambda_{13}, \lambda_{14}] \quad (9)$$

```
> subs(SOL,linalg[hessian](Lag, X)) ; Hess1 := convert(%,Matrix):
```

[[-329.6921814, 329.6921814, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], **(10)**

[329.6921814, -600.1388100, 270.4466286, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
],

[0, 270.4466286, -486.0265726, 215.5799440, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
],

[0, 0, 215.5799440, -384.9816397, 169.4016957, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
],

[0, 0, 0, 169.4016957, -310.1427544, 140.7410587, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
],

[0, 0, 0, 0, 140.7410587, -281.4821174, 140.7410587, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
],
[0, 0, 0, 0, 0, 140.7410587, -310.1427544, 169.4016957, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
],
[0, 0, 0, 0, 0, 0, 169.4016957, -384.9816397, 215.5799440, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
],
[0, 0, 0, 0, 0, 0, 0, 215.5799440, -486.0265726, 270.4466286, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
],
[0, 0, 0, 0, 0, 0, 0, 0, 270.4466286, -600.1388100, 329.6921814, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 329.6921814, -329.6921814, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -329.6921814, 329.6921814, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 329.6921814, -600.1388100, 270.4466286, 0, 0, 0, 0, 0, 0, 0, 0
],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 270.4466286, -486.0265726, 215.5799440, 0, 0, 0, 0, 0, 0, 0
],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 215.5799440, -384.9816397, 169.4016957, 0, 0, 0, 0, 0, 0
],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 169.4016957, -310.1427544, 140.7410587, 0, 0, 0, 0, 0
],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 140.7410587, -281.4821174, 140.7410587, 0, 0, 0, 0
],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 140.7410587, -310.1427544, 169.4016957, 0, 0, 0
],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 169.4016957, -384.9816397, 215.5799440, 0, 0
],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 215.5799440, -486.0265726, 270.4466286, 0
],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 270.4466286, -600.1388100, 329.6921814
],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 329.6921814, -329.6921814]]

```
> Eigenvalues(Hess1) : convert(%,list);
```

[-961.199148743534 + 0. I, -961.009746283056 + 0. I, -638.111176448330 + 0. I, **(11)**
-617.411185272176 + 0. I, -471.417696948675 + 0. I, -369.098318163323 + 0. I,
-252.758355643212 + 0. I, -149.461573483036 + 0. I, $4.35853473103464\ 10^{-14}$ + 0. I,
-14.0011348984091 + 0. I, -68.9776977162479 + 0. I, -961.199148743534 + 0. I,
-961.009746283056 + 0. I, -638.111176448330 + 0. I, -617.411185272176 + 0. I,

$-471.417696948675 + 0.\ I, -369.098318163323 + 0.\ I, -252.758355643212 + 0.\ I,$
$-149.461573483036 + 0.\ I, 4.35853473103464\ 10^{-14} + 0.\ I, -14.0011348984091 + 0.\ I,$
$-68.9776977162479 + 0.\ I]$

```
> H1 := <seq( subs(SOL,linalg[grad](h||k, X)),k=1..N+4)>;
```

$$H1 := \left[\begin{array}{c} \text{14 x 22 Matrix} \\ \text{Data Type: anything} \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array}\right] \quad (12)$$

```
> Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8 := op(NullSpace(H1));
```

$$Z1, Z2, Z3, Z4, Z5, Z6, Z7, Z8 := \left[\begin{array}{c} 1 .. 22\ \text{Vector}_{column} \\ \text{Data Type: } float_8 \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array}\right], \left[\begin{array}{c} 1 .. 22\ \text{Vector}_{column} \\ \text{Data Type: } float_8 \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array}\right], \quad (13)$$
$$\left[\begin{array}{c} 1 .. 22\ \text{Vector}_{column} \\ \text{Data Type: } float_8 \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array}\right], \left[\begin{array}{c} 1 .. 22\ \text{Vector}_{column} \\ \text{Data Type: } float_8 \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array}\right], \left[\begin{array}{c} 1 .. 22\ \text{Vector}_{column} \\ \text{Data Type: } float_8 \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array}\right],$$
$$\left[\begin{array}{c} 1 .. 22\ \text{Vector}_{column} \\ \text{Data Type: } float_8 \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array}\right], \left[\begin{array}{c} 1 .. 22\ \text{Vector}_{column} \\ \text{Data Type: } float_8 \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array}\right], \left[\begin{array}{c} 1 .. 22\ \text{Vector}_{column} \\ \text{Data Type: } float_8 \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array}\right]$$

```
> ZKer := add(alpha||j*Z||j,j=1..8) ;
```

$$ZKer := \left[\begin{array}{c} 1 .. 22\ \text{Vector}_{column} \\ \text{Data Type: anything} \\ \text{Storage: rectangular} \\ \text{Order: Fortran\_order} \end{array}\right] \quad (14)$$

```
> AA := simplify(Transpose(ZKer).Hess1.ZKer) ;
```

$$AA := -262.4677771\,\alpha5^2 + 293.7092844\,\alpha5\,\alpha6 - 677.1031767\,\alpha6^2 - 112.0932419\,\alpha3\,\alpha1 \quad (15)$$
$$- 291.3209725\,\alpha3^2 - 407.9199304\,\alpha1^2 - 169.3136083\,\alpha8\,\alpha1 + 152.1299411\,\alpha8\,\alpha2$$
$$+ 99.11948622\,\alpha8\,\alpha3 + 98.69552444\,\alpha8\,\alpha4 - 252.5877833\,\alpha8\,\alpha7 - 663.4892891\,\alpha8^2$$
$$- 221.6201262\,\alpha7\,\alpha1 + 1.748688161\,\alpha7\,\alpha2 - 46.23683186\,\alpha7\,\alpha3$$
$$- 141.3957137\,\alpha7\,\alpha4 - 664.3338455\,\alpha7^2 - 332.2438261\,\alpha5\,\alpha1 - 30.48949608\,\alpha5\,\alpha2$$

$- 148.6285587\, \alpha 5\, \alpha 3 - 24.12043228\, \alpha 5\, \alpha 4 - 24.03730404\, \alpha 5\, \alpha 7$

$+ 325.2085263\, \alpha 5\, \alpha 8 + 46.25407601\, \alpha 6\, \alpha 1 + 5.043604046\, \alpha 6\, \alpha 2$

$+ 145.8483089\, \alpha 6\, \alpha 3 + 260.2374815\, \alpha 6\, \alpha 4 + 104.7798940\, \alpha 6\, \alpha 7$

$- 102.0160091\, \alpha 6\, \alpha 8 - 148.3896251\, \alpha 4\, \alpha 1 - 89.42857812\, \alpha 4\, \alpha 2$

$+ 163.1090097\, \alpha 4\, \alpha 3 - 294.9975070\, \alpha 4^2 + 273.1464687\, \alpha 2\, \alpha 1 - 365.6897934\, \alpha 2^2$

$+ 230.6534102\, \alpha 2\, \alpha 3$

Extract the matrix associated to alphas

```
> HessReduced := Matrix(8,8) ;
```

$$HessReduced := \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad (16)$$

```
> for i from 1 to 8 do
    for j from 1 to 8 do
      HessReduced[i,j] := diff(AA,alpha||i,alpha||j)/2 ;
    end :
  end :
> HessReduced;
```

[[-407.9199304, 136.5732344, -56.04662095, -74.19481255, -166.1219130, 23.12703800, -110.8100631, -84.65680415], **(17)**
[136.5732344, -365.6897934, 115.3267051, -44.71428906, -15.24474804, 2.521802023, 0.8743440805, 76.06497055],
[-56.04662095, 115.3267051, -291.3209725, 81.55450485, -74.31427935, 72.92415445, -23.11841593, 49.55974311],
[-74.19481255, -44.71428906, 81.55450485, -294.9975070, -12.06021614, 130.1187408, -70.69785685, 49.34776222],
[-166.1219130, -15.24474804, -74.31427935, -12.06021614, -262.4677771, 146.8546422, -12.01865202, 162.6042632],
[23.12703800, 2.521802023, 72.92415445, 130.1187408, 146.8546422, -677.1031765, 52.38994700, -51.00800455],
[-110.8100631, 0.8743440805, -23.11841593, -70.69785685, -12.01865202, 52.38994700, -664.3338455, -126.2938916],
[-84.65680415, 76.06497055, 49.55974311, 49.34776222, 162.6042632, -51.00800455, -126.2938916, -663.4892890]]

Use Sylvester theorem to check if is SPD

```
> for k from 1 to 8 do
    Determinant(-HessReduced[1..k,1..k]) ;
  end;
```

$$407.919930400000$$

$$1.30519906717241\ 10^{5}$$

$$3.32145653097046\ 10^{7}$$

$$8.19431058640211\ 10^{9}$$

$$1.29089730509999\ 10^{12}$$

$$6.14548621480045\ 10^{14}$$

$$3.70451999980361\ 10^{17}$$

$$1.34090499423090\ 10^{20} \quad \textbf{(18)}$$