Equazioni Eulero-Lagrange

Le equazioni di Eulero-Lagrange si basano sul principio di Hamilton(principio di minimizzazione) in cui, note le espressioni dell'energia cinetica e potenziale, si calcolano le equazioni del moto e quindi permette di simulare il comportamento del sistema.

Sia:

$$T := \text{energica cinetica}, U := \text{energia potenziale}$$

Si definisce l'**azione** come $\int_{\mathrm{ti}}^{t_f} \mathcal{L} \partial t$ in cui $\mathcal{L} := \text{Lagrangiana} = T - U$.

Supponiamo di voler passare da $q(t_i) \rightarrow q(t_f)$:

Si possono percorrere innumerevoli traiettorie, tuttavia dimostra che la traiettoria a costo minimo in (t_i, t_f) coincide con $\dot{q}^*(t), \text{cioè un punto di stazionarietà dell'integrale } d'\text{azione}.$

N.B.:

- Si definisce **punto stazionario** un punto $f(x)$ t.c. $\frac{\partial f}{\partial x} = 0$.

Inoltre ,effettuando lo sviluppo di Taylor della fuzione $y = f(x)$ che congiunge i punti dell'ipotesi.Si definisce la variazione $f(x) - f(x_0) = \partial y = J(x_0)(x - x_0) + \ldots$ in cui $(x - x_0) = \partial x$. Quindi:

$$\partial y = J(x) \partial x$$

Si definisce un **punto x^* di stazionarietà** se $\partial y = 0 \forall \partial x$.

In altre parole, un punto è di stazionarietà se considere una variazione nel codominio della funzione, ottengo una variazione nulla del dominio:

$$J(x^*) = 0$$

Il principio di Hamilton afferma che il sistema si muove in maniera tale che:

$$\partial J = 0 \qquad \forall q$$

Da cui si ottengono le equazioni di Eulero-Lagrange.Esse tengono conto del tipo di sistema che si sta considerando:

-Sistemi conservativi(assenza di forze esterne e forze d'attrito):

$$\frac{\mathrm{d}}{\mathrm{dt}} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = 0$$

-Sistemi non conservativi(presenza di forze esterne):

$$\frac{\mathrm{d}}{\mathrm{dt}} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = u^T$$

Il vettore u^T rappresenta le forze esterne agenti lungo la direzione di q.

In generale, per ottenere l'equazione di un qualsiasi moto, occorre considerare anche la presenza delle forze di attrito. Si ottiene:

$$\frac{\mathrm{d}}{\mathrm{dt}} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} + \frac{\partial \mathbb{F}}{\partial \dot{q}} = u^T$$

In cui $\mathbb{F} = \frac{1}{2}\dot{q}^T F \dot{q}$, $F \succeq 0$.

```
(%i1) inverseLaplace(SI,theta):=block([res,M,MC,aC,b],
                                M:SI,
                                MC:SI,
                                for i:1 thru 3 do(
                                  for j:1 thru 3 do
                                      (
                                        aC:M[i,j],
                                        b:ilt(aC,s,theta),
                                        MC[i,j]:b
                                       )
                                   ),
                                res:MC

                             )
```

(%o1) $\mathrm{inverseLaplace}(\mathrm{SI}, \vartheta) := \textbf{block}\ ([\mathrm{res}, M, \mathrm{MC}, \mathrm{aC}, b], M{:}\,\mathrm{SI}, \mathrm{MC}{:}\,\mathrm{SI},$
$\textbf{for}\ i\ \textbf{thru}\ 3\ \textbf{do}\ \textbf{for}\ j\ \textbf{thru}\ 3\ \textbf{do}\ (\mathrm{aC}{:}\,M_{i,j}, b{:}\,\mathrm{ilt}(\mathrm{aC}, s, \vartheta), \mathrm{MC}_{i,j}{:}\,b), \mathrm{res}{:}\,\mathrm{MC})$

```
(%i2) rotLaplace(k,theta):=block([res,S,I],
                            S:ident(3),
                            I:ident(3),
                          for i:1 thru 3 do
                            (
                            for j:1 thru 3 do
                               (
                                 if i=j
                                    then S[i][j]:0
                                 elseif j>i
                                    then (
                                   temp:(-1)^(j-i)*k[3-remainder(i+j,3)],
                                          S[i][j]:temp,
                                          S[j][i]:-temp

                                           )
                                 )
                             ),
                            res:inverseLaplace(invert(s*I-S),theta)


                          )
```

(%o2) $\mathrm{rotLaplace}(k, \vartheta) := \textbf{block}\ ([\mathrm{res}, S, I], S{:}\,\mathrm{ident}(3), I{:}\,\mathrm{ident}(3),$
$\textbf{for}\ i\ \textbf{thru}\ 3\ \textbf{do}\ \textbf{for}\ j\ \textbf{thru}\ 3\ \textbf{do}\ \textbf{if}\ i = j\ \textbf{then}\ (S_i)_j{:}\,0\ \textbf{elseif}\ j > i\ \textbf{then}\ (\mathrm{temp}{:}$
$(-1)^{j-i} k_{3-\mathrm{remainder}(i+j,3)}, (S_i)_j{:}\,\mathrm{temp}, (S_j)_i{:}\,-\mathrm{temp}), \mathrm{res}{:}\,\mathrm{inverseLaplace}(\mathrm{invert}(s\,I - S), \vartheta))$

```
(%i3) Av(v,theta,d):=block([res,Trot,row,Atemp,A],
                            Trot:rotLaplace(v,theta),
                            row:matrix([0,0,0,1]),
                            Atemp:addcol(Trot,d*transpose(v)),
                            A:addrow(Atemp,row),
                            res:A
                            )
```

(%o3) $\mathrm{Av}(v,\vartheta,d) := \textbf{block}\ ([\mathrm{res}, \mathrm{Trot}, \mathrm{row}, \mathrm{Atemp}, A], \mathrm{Trot}\colon \mathrm{rotLaplace}(v,\vartheta), \mathrm{row}\colon \begin{pmatrix} 0 & 0 & 0 & 1 \end{pmatrix},$ $\mathrm{Atemp}\colon \mathrm{addcol}(\mathrm{Trot}, d\,\mathrm{transpose}(v)), A\colon \mathrm{addrow}(\mathrm{Atemp}, \mathrm{row}), \mathrm{res}\colon A)$

```
(%i4) Q(theta,d,alpha,a):=block([res,tempMat,Qtrasf],
                              tempMat:Av([0,0,1],theta,d).Av([1,0,0],alpha,a),
                              Qtrasf:zeromatrix(4,4),
                              for i:1 thru 4 do
                         (
                         for j:1 thru 4 do
                            (
                              Qtrasf[i][j]:trigreduce(tempMat[i][j])
                                        )
                            ),

                              res:Qtrasf
                              )
```

(%o4) $Q(\vartheta,d,\alpha,a) := \textbf{block}\ ([\mathrm{res}, \mathrm{tempMat}, \mathrm{Qtrasf}], \mathrm{tempMat}\colon \mathrm{Av}([0,0,1],\vartheta,d)\cdot \mathrm{Av}([1,0,0],\alpha,a),$ $\mathrm{Qtrasf}\colon \mathrm{zeromatrix}(4,4), \textbf{for } i \textbf{ thru } 4 \textbf{ do for } j \textbf{ thru } 4 \textbf{ do } (\mathrm{Qtrasf}_i)_j\colon \mathrm{trigreduce}((\mathrm{tempMat}_i)_j), \mathrm{res}\colon$ $\mathrm{Qtrasf})$

```
(%i5) Qdirect(DH):=block([res,Q,Qtemp],
                       Q:[Q(DH[1][1],DH[1][2],DH[1][3],DH[1][4])],
                       for i:2 thru length(DH) do(

                            Qtemp:Q(DH[i][1],DH[i][2],DH[i][3],DH[i][4]),

                            Q:append(Q,[trigsimp(trigreduce(trigexpand(Q[i-
      1].Qtemp)))])


                           ),
                        res:Q)
```

(%o5) $\mathrm{Qdirect}(\mathrm{DH}) := \textbf{block}\ ([\mathrm{res}, Q, \mathrm{Qtemp}], Q\colon [Q((\mathrm{DH}_1)_1, (\mathrm{DH}_1)_2, (\mathrm{DH}_1)_3, (\mathrm{DH}_1)_4)],$ $\textbf{for } i \textbf{ from } 2 \textbf{ thru } \mathrm{length}(\mathrm{DH}) \textbf{ do } (\mathrm{Qtemp}\colon Q((\mathrm{DH}_i)_1, (\mathrm{DH}_i)_2, (\mathrm{DH}_i)_3, (\mathrm{DH}_i)_4), Q\colon \mathrm{append}(Q,$ $[\mathrm{trigsimp}(\mathrm{trigreduce}(\mathrm{trigexpand}(Q_{i-1}\cdot \mathrm{Qtemp})))])), \mathrm{res}\colon Q)$

```
(%i6) Qbc(Q,bc,dist):=block([traslBC,Qcap],
                    Qcap:[],
                    ex:matrix([1],[0],[0]), ez:matrix([0],[0],[1]),
                    for j:1 thru length(Q) do(
                             traslBC:addrow(addcol(ident(3),dist[j]),[0,0,0,1]),
                             Qcap:append(Qcap,[trigsimp(Q[j].traslBC)])

                             ),
                    Qcap
                    )
```

(%o6) $\mathrm{Qbc}(Q,\mathrm{bc},\mathrm{dist}) := \textbf{block}\left([\mathrm{traslBC}, \mathrm{Qcap}], \mathrm{Qcap}\colon [], \mathrm{ex}\colon \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \mathrm{ez}\colon \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix},\right.$ $\textbf{for } j \textbf{ thru } \mathrm{length}(Q) \textbf{ do } (\mathrm{traslBC}\colon \mathrm{addrow}(\mathrm{addcol}(\mathrm{ident}(3), \mathrm{dist}_j), [0,0,0,1]), \mathrm{Qcap}\colon \mathrm{append}(\mathrm{Qcap},$ $\left.[\mathrm{trigsimp}(Q_j\cdot \mathrm{traslBC})])), \mathrm{Qcap}\right)$

(%i7)
```
inerzia(j):=block(
[II],
II:matrix([alpha[xx][i],alpha[xy][i],alpha[xz][i]],
          [alpha[xy][i],alpha[yy][i],alpha[yz][i]],
          [alpha[xz][i],alpha[yz][i],alpha[zz][i]]),
II:subst(j, i, II),
return(II)
)
```

(%o7) $$\mathrm{inerzia}(j):=\mathbf{block}\left([\mathrm{II}],\mathrm{II}:\begin{pmatrix}(\alpha_{\mathrm{xx}})_i & (\alpha_{\mathrm{xy}})_i & (\alpha_{\mathrm{xz}})_i\\ (\alpha_{\mathrm{xy}})_i & (\alpha_{\mathrm{yy}})_i & (\alpha_{\mathrm{yz}})_i\\ (\alpha_{\mathrm{xz}})_i & (\alpha_{\mathrm{yz}})_i & (\alpha_{\mathrm{zz}})_i\end{pmatrix},\mathrm{II}:\mathrm{subst}(j,i,\mathrm{II}),\mathrm{return}(\mathrm{II})\right)$$

(%i8)
```
massa(k):=M[k];
```

(%o8) $\mathrm{massa}(k):=M_k$

(%i9)
```
ek(DH,dist):=block([Q,Qcap,I,wtemp,w,Si,Tatemp,Ta,Tbtemp,Tb,d,dd,Qend,B,
Btemp,T,Tot,Btot,res],
                   I:[],w:[],Ta:[],Tb:[],B:[],T:[],Ttot:0,
                   depends([q],t),
                   Q:Qdirect(DH),

                   Qcap:Qbc(Q,DH,dist),

                  for i:1 thru length(Qcap) do( I:append(I,[inerzia(i)]),
                       R:matrix([Qcap[i][1][1],Qcap[i][1][2],
Qcap[i][1][3]],[Qcap[i][2][1],Qcap[i][2][2],Qcap[i][2][3]],[Qcap[i][3][1],
Qcap[i][3][2],Qcap[i][3][3]]),
                      dR:diff(R,t),
                     /* for j:1 thru length(DH) do(
                          dR:subst('diff(q[j],t)=omega[j],dR)),*/
                      Sw:dR.transpose(R),
                      wtemp:matrix([Sw[3][2]],[Sw[1][3]],[Sw[2][1]]),
                      w:append(w,[trigreduce(expand(wtemp))]),
                 Tatemp:(1/2)*transpose(wtemp).R.I[i].transpose(R).wtemp,
                 Tatemp:trigsimp(trigreduce(trigexpand(Tatemp))),
                  Ta:append(Ta,[Tatemp]),
                  d:matrix([Qcap[i][1][4]],[Qcap[i][2][4]],[Q[i][3][4]]),
                  dd:diff(d,t),
                 /* for j:1 thru length(DH) do(dd:subst('diff(q[j],
t)=omega[j],dd)),*/
                  Tbtemp:(massa(i)/
2)*trigsimp(trigreduce(trigexpand(transpose(dd).dd))),
                  Tb:append(Tb,[Tbtemp]),
                  T:append(T,[trigreduce(Tatemp+Tbtemp)])),
                  for i:1 thru length(DH) do(
                    Ttot:T[i]+Ttot
                  ),
                  Ttot
                   )
```

(%o9) $\mathrm{ek}(\mathrm{DH},\mathrm{dist}):=\mathbf{block}\Big([Q,\mathrm{Qcap},I,\mathrm{wtemp},w,\mathrm{Si},\mathrm{Tatemp},\mathrm{Ta},\mathrm{Tbtemp},\mathrm{Tb},d,\mathrm{dd},\mathrm{Qend},B,$

$\mathrm{Btemp},T,\mathrm{Tot},\mathrm{Btot},\mathrm{res}],I:[],w:[],\mathrm{Ta}:[],\mathrm{Tb}:[],B:[],T:[],\mathrm{Ttot}:0,\mathrm{depends}([q],t),Q:\mathrm{Qdirect}(\mathrm{DH}),$

$\text{Qcap: Qbc}(Q, \text{DH}, \text{dist}), \textbf{for } i \textbf{ thru } \text{length}(\text{Qcap}) \textbf{ do } \Bigg(I\text{: append}(I, [\text{inerzia}(i)]), R\text{:}$
$\begin{pmatrix} ((\text{Qcap}_i)_1)_1 & ((\text{Qcap}_i)_1)_2 & ((\text{Qcap}_i)_1)_3 \\ ((\text{Qcap}_i)_2)_1 & ((\text{Qcap}_i)_2)_2 & ((\text{Qcap}_i)_2)_3 \\ ((\text{Qcap}_i)_3)_1 & ((\text{Qcap}_i)_3)_2 & ((\text{Qcap}_i)_3)_3 \end{pmatrix}, \text{dR: diff}(R, t), \text{Sw: dR} \cdot \text{transpose}(R), \text{wtemp:}$
$\begin{pmatrix} (\text{Sw}_3)_2 \\ (\text{Sw}_1)_3 \\ (\text{Sw}_2)_1 \end{pmatrix}, w\text{: append}(w, [\text{trigreduce}(\text{expand}(\text{wtemp}))]), \text{Tatemp: } \frac{1}{2} \text{transpose}(\text{wtemp}) \cdot R \cdot I_i \cdot$
$\text{transpose}(R) \cdot \text{wtemp}, \text{Tatemp: trigsimp}(\text{trigreduce}(\text{trigexpand}(\text{Tatemp}))), \text{Ta: append}(\text{Ta},$
$[\text{Tatemp}]), d\text{:} \begin{pmatrix} ((\text{Qcap}_i)_1)_4 \\ ((\text{Qcap}_i)_2)_4 \\ ((Q_i)_3)_4 \end{pmatrix}, \text{dd: diff}(d, t), \text{Tbtemp:}$
$\frac{\text{massa}(i)}{2} \text{trigsimp}(\text{trigreduce}(\text{trigexpand}(\text{transpose}(\text{dd}) \cdot \text{dd}))), \text{Tb: append}(\text{Tb}, [\text{Tbtemp}]), T\text{:}$
$\text{append}(T, [\text{trigreduce}(\text{Tatemp} + \text{Tbtemp})]) \Bigg), \textbf{for } i \textbf{ thru } \text{length}(\text{DH}) \textbf{ do } \text{Ttot: } T_i + \text{Ttot}, \text{Ttot} \Bigg)$

```
(%i10) ep(DH,dist):=block([Q,Qcap,g,U,Utemp,dTemp,prev,Utot],
                     Q:[], Qcap:[],U:[],Utot:zeromatrix(3,3),Utot:0,
                      depends([q,omega],t),
                     g:10*matrix([0],[0],[1]),
                     prev:ident(4),
                     Q:Qdirect(DH),
                      Qcap:Qbc(Q,DH,dist),

                        for i:1 thru length(Qcap) do(

                         dTemp:matrix([Qcap[i][1][4]],[Qcap[i][2][4]],
                                       [Qcap[i][3][4]]),

                         Utemp:M[i]*transpose(g).dTemp,
                         U:append(U,[Utemp])

                       ),
                        for i:1 thru length(U) do(
                            Utot:Utot+U[i]
                        ),
                        ratsimp(trigsimp(trigreduce(trigexpand(Utot))))
                        );
```

(%o10) $\text{ep}(\text{DH}, \text{dist}) := \textbf{block}\Bigg([Q, \text{Qcap}, g, U, \text{Utemp}, \text{dTemp}, \text{prev}, \text{Utot}], Q\text{: } [], \text{Qcap: } [], U\text{: } [],$
$\text{Utot: zeromatrix}(3, 3), \text{Utot: } 0, \text{depends}([q, \omega], t), g\text{: } 10 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \text{prev: ident}(4), Q\text{: Qdirect}(\text{DH}),$
$\text{Qcap: Qbc}(Q, \text{DH}, \text{dist}), \textbf{for } i \textbf{ thru } \text{length}(\text{Qcap}) \textbf{ do } \Bigg(\text{dTemp: } \begin{pmatrix} ((\text{Qcap}_i)_1)_4 \\ ((\text{Qcap}_i)_2)_4 \\ ((\text{Qcap}_i)_3)_4 \end{pmatrix}, \text{Utemp:}$
$M_i \,\text{transpose}(g) \cdot \text{dTemp}, U\text{: append}(U, [\text{Utemp}]) \Bigg), \textbf{for } i \textbf{ thru } \text{length}(U) \textbf{ do } \text{Utot: Utot} + U_i,$
$\text{ratsimp}(\text{trigsimp}(\text{trigreduce}(\text{trigexpand}(\text{Utot})))) \Bigg)$

```
(%i11) eel(dh,f,u,dist):=block([T,U,L,dLq,dLqt,dLqp,F,eq,eqi,vel],
                 print(q[i][i],"Indica la derivata i-esima di",q[i]),
                 T:0,U:0,eq:zeromatrix(length(dh),1),vel:zeromatrix(length(dh),1),
                 dLq:zeromatrix(length(dh),1),dLqt:zeromatrix(length(dh),1),
                 dLqp:zeromatrix(length(dh),1),dF:zeromatrix(length(dh),1),
                 eq:zeromatrix(length(dh),1),depends([q],t),
                 for i:1 thru length(dh) do(vel[i][1]:diff(q[i],t)),
                 T:ek(dh,dist), U:ep(dh,dist),
                 if length(dh)=1 then(F:(1/2)*(transpose(vel).vel)*f)else
                 (F:(1/2)*expand(transpose(vel).f.vel)),
                 L:trigsimp(trigreduce(trigexpand(T-U))),
                 for i:1 thru length(dh) do
                    (dLq[i][1]:diff(L,'diff(q[i],t)),
                     dLqt[i][1]:diff(dLq[i][1],t),
                     dLqp[i][1]:diff(L,q[i]),
                     dF[i][1]:expand(diff(F,'diff(q[i],t))),
                     eq[i][1]:dLqt[i][1]-dLqp[i][1]+dF[i][1]-u[i]),
                 for i:1 thru length(dh) do
                    (T:subst('diff(q[i],t)=q[i][1],T),
                     T:subst('diff(diff(q[i],t),t)=q[i][2],T),
                     F:subst('diff(q[i],t)=q[i][1],F),
                     F:subst('diff(diff(q[i],t),t)=q[i][2],F),
                     U:subst('diff(q[i],t)=q[i][1],U),
                     U:subst('diff(diff(q[i],t),t)=q[i][2],U),
                     L:subst('diff(q[i],t)=q[i][1],L),
                     L:subst('diff(diff(q[i],t),t)=q[i][2],L),
                     dLq:subst('diff(q[i],t)=q[i][1],dLq),
                     dLq:subst('diff(diff(q[i],t),t)=q[i][2],dLq),
                     dLqt:subst('diff(q[i],t)=q[i][1],dLqt),
                     dLqt:subst('diff(diff(q[i],t),t)=q[i][2],dLqt),
                     dLqp:subst('diff(q[i],t)=q[i][1],dLqp),
                     dLqp:subst('diff(diff(q[i],t),t)=q[i][2],dLqp),
                     dF:subst('diff(q[i],t)=q[i][1],dF),
                     dF:subst('diff(diff(q[i],t),t)=q[i][2],dF),
                     eq:subst('diff(q[i],t)=q[i][1],eq),
                     eq:subst('diff(diff(q[i],t),t)=q[i][2],eq)),
                     print("Energia cinetica T=",ratsimp(expand(T))),
                     print("Energia potenziale U=",ratsimp(expand(U))),
                     print("Forze di attrito F=",F),
                     print("Forze esterne u=",u),
                     print("Lagrangiana L=",ratsimp(trigreduce(expand(L)))),
                     print("dL/dq' = ", ratsimp(trigreduce(expand(dLq)))),
                     print("d/dt dL/dq' = ", ratsimp(trigreduce(expand(dLqt)))),
                     print("dL/dq = ",ratsimp(trigreduce(expand(dLqp)))),
                     print("dF/dq' = ",ratsimp(dF)),
                     for i:1 thru length(dh) do(
                          print("Equazione eulero lagrange",i),
                                print(ratsimp(eq[i][1][1]),"=0")
                     ));
```

(%o11) $\mathrm{eel}(\mathrm{dh}, f, u, \mathrm{dist}) := \mathbf{block}\Big([T, U, L, \mathrm{dLq}, \mathrm{dLqt}, \mathrm{dLqp}, F, \mathrm{eq}, \mathrm{eqi}, \mathrm{vel}], \mathrm{print}((q_i)_i, \text{Indica la derivata i-esima di }, q_i), T\colon 0, U\colon 0, \mathrm{eq}\colon \mathrm{zeromatrix}(\mathrm{length}(\mathrm{dh}), 1), \mathrm{vel}\colon \mathrm{zeromatrix}(\mathrm{length}(\mathrm{dh}), 1), \mathrm{dLq}\colon \mathrm{zeromatrix}(\mathrm{length}(\mathrm{dh}), 1), \mathrm{dLqt}\colon \mathrm{zeromatrix}(\mathrm{length}(\mathrm{dh}), 1), \mathrm{dLqp}\colon \mathrm{zeromatrix}(\mathrm{length}(\mathrm{dh}), 1), \mathrm{dF}\colon \mathrm{zeromatrix}(\mathrm{length}(\mathrm{dh}), 1), \mathrm{eq}\colon \mathrm{zeromatrix}(\mathrm{length}(\mathrm{dh}), 1), \mathrm{depends}([q], t),$
$\mathbf{for}\ i\ \mathbf{thru}\ \mathrm{length}(\mathrm{dh})\ \mathbf{do}\ (\mathrm{vel}_i)_1\colon \mathrm{diff}(q_i, t), T\colon \mathrm{ek}(\mathrm{dh}, \mathrm{dist}), U\colon \mathrm{ep}(\mathrm{dh}, \mathrm{dist}), \mathbf{if}\ \mathrm{length}(\mathrm{dh}) =$

1 **then** F: $\frac{1}{2}$ transpose(vel) $\cdot$ vel f **else** F: $\frac{1}{2}$ expand(transpose(vel) $\cdot f \cdot$ vel), L: trigsimp(trigreduce(trigexpand($T-U$))), **for** i **thru** length(dh) **do** $\left((\mathrm{dLq}_i)_1\text{: diff}\left(L, \frac{1}{\mathrm{mtimes}()}\, q_i\right), (\mathrm{dLqt}_i)_1\text{: diff}((\mathrm{dLq}_i)_1, t), (\mathrm{dLqp}_i)_1\text{: diff}(L, q_i), (\mathrm{dF}_i)_1\text{: expand}\left(\mathrm{diff}\left(F, \frac{1}{\mathrm{mtimes}()}\, q_i\right)\right), (\mathrm{eq}_i)_1\text{: } (\mathrm{dLqt}_i)_1 - (\mathrm{dLqp}_i)_1 + (\mathrm{dF}_i)_1 - u_i\right)$, **for** i **thru** length(dh) **do** $\Big(T\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, q_i = (q_i)_1, T\right), T\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, \mathrm{diff}(q_i, t) = (q_i)_2, T\right), F\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, q_i = (q_i)_1, F\right), F\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, \mathrm{diff}(q_i, t) = (q_i)_2, F\right), U\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, q_i = (q_i)_1, U\right), U\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, \mathrm{diff}(q_i, t) = (q_i)_2, U\right), L\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, q_i = (q_i)_1, L\right), L\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, \mathrm{diff}(q_i, t) = (q_i)_2, L\right), \mathrm{dLq}\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, q_i = (q_i)_1, \mathrm{dLq}\right), \mathrm{dLq}\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, \mathrm{diff}(q_i, t) = (q_i)_2, \mathrm{dLq}\right), \mathrm{dLqt}\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, q_i = (q_i)_1, \mathrm{dLqt}\right), \mathrm{dLqt}\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, \mathrm{diff}(q_i, t) = (q_i)_2, \mathrm{dLqt}\right), \mathrm{dLqp}\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, q_i = (q_i)_1, \mathrm{dLqp}\right), \mathrm{dLqp}\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, \mathrm{diff}(q_i, t) = (q_i)_2, \mathrm{dLqp}\right), \mathrm{dF}\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, q_i = (q_i)_1, \mathrm{dF}\right), \mathrm{dF}\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, \mathrm{diff}(q_i, t) = (q_i)_2, \mathrm{dF}\right), \mathrm{eq}\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, q_i = (q_i)_1, \mathrm{eq}\right), \mathrm{eq}\text{: subst}\left(\frac{1}{\mathrm{mtimes}()}\, \mathrm{diff}(q_i, t) = (q_i)_2, \mathrm{eq}\right)\Big)$, print(Energia cinetica T= , ratsimp(expand(T))), print(Energia potenziale U= , ratsimp(expand(U))), print(Forze di attrito F= , F), print(Forze esterne u= , u), print(Lagrangiana L= , ratsimp(trigreduce(expand(L)))), print(dL/dq' = , ratsimp(trigreduce(expand(dLq)))), print(d/dt dL/dq' = , ratsimp(trigreduce(expand(dLqt)))), print(dL/dq = , ratsimp(trigreduce(expand(dLqp)))), print(dF/dq' = , ratsimp(dF)),
for i **thru** length(dh) **do** (print(Equazione eulero lagrange , i), print(ratsimp((($\mathrm{eq}_i)_1)_1$), =0)))

2DOF

(%i12) `dueDof:[[q[1],0,0,L[1]],[q[2],0,0,L[2]]];`

(%o12) $[[q_1, 0, 0, L_1], [q_2, 0, 0, L_2]]$

(%i13) `u:matrix([u[1]],[u[2]])`

(%o13) $\begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$

(%i14) `F:matrix([K[11],0],[0,K[22]]);`

(%o14) $\begin{pmatrix} K_{11} & 0 \\ 0 & K_{22} \end{pmatrix}$

(%i15) `distance:[matrix([-L[1]/2],[0],[0]),matrix([-L[2]/2],[0],[0])];`

(%o15) $\left[\begin{pmatrix} -\frac{L_1}{2} \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} -\frac{L_2}{2} \\ 0 \\ 0 \end{pmatrix}\right]$

(%i16) `eel(dueDof,F,u,distance);`

$(q_i)_i$ Indica la derivata i-esima di q_i

Energia cinetica T= $((4\,(q_1)_1\,(q_2)_1+4\,(q_1)_1^2)\,L_1\,L_2\,M_2\cos(q_2)+(4\,(q_2)_1^2+8\,(q_1)_1\,(q_2)_1+4\,(q_1)_1^2)\,(\alpha_{zz})_2+(((q_2)_1^2+2\,(q_1)_1\,(q_2)_1+(q_1)_1^2)\,L_2^2+4\,(q_1)_1^2\,L_1^2)\,M_2+4\,(q_1)_1^2\,(\alpha_{zz})_1+(q_1)_1^2\,L_1^2\,M_1)/8$

Energia potenziale U= 0

Forze di attrito F= $\dfrac{(q_2)_1^2\,K_{22}+(q_1)_1^2\,K_{11}}{2}$

Forze esterne u= $\begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$

Lagrangiana L= $((4\,(q_1)_1\,(q_2)_1+4\,(q_1)_1^2)\,L_1\,L_2\,M_2\cos(q_2)+(4\,(q_2)_1^2+8\,(q_1)_1\,(q_2)_1+4\,(q_1)_1^2)\,(\alpha_{zz})_2+(((q_2)_1^2+2\,(q_1)_1\,(q_2)_1+(q_1)_1^2)\,L_2^2+4\,(q_1)_1^2\,L_1^2)\,M_2+4\,(q_1)_1^2\,(\alpha_{zz})_1+(q_1)_1^2\,L_1^2\,M_1)/8$

dL/dq' = $\left(((2\,(q_2)_1+4\,(q_1)_1)\,L_1\,L_2\,M_2\cos(q_2)+(4\,(q_2)_1+4\,(q_1)_1)\,(\alpha_{zz})_2+(((q_2)_1+(q_1)_1)\,L_2^2+4\,(q_1)_1\,L_1^2)\,M_2+4\,(q_1)_1\,(\alpha_{zz})_1+(q_1)_1\,L_1^2\,M_1)/4;\ \frac{2\,(q_1)_1\,L_1\,L_2\,M_2\cos(q_2)+(4\,(q_2)_1+4\,(q_1)_1)\,(\alpha_{zz})_2+((q_2)_1+(q_1)_1)\,L_2^2\,M_2}{4}\right)$

d/dt dL/dq' = $(-((2\,(q_2)_1^2+4\,(q_1)_1\,(q_2)_1)\,L_1\,L_2\,M_2\sin(q_2)+(-2\,L_1\,(q_2)_2-4\,L_1\,(q_1)_2)\,L_2\,M_2\cos(q_2)+(-4\,(q_2)_2-4\,(q_1)_2)\,(\alpha_{zz})_2+((-(q_2)_2-(q_1)_2)\,L_2^2-4\,L_1^2\,(q_1)_2)\,M_2+(-4\,(\alpha_{zz})_1-L_1^2\,M_1)\,(q_1)_2)/4;-(2\,(q_1)_1\,(q_2)_1\,L_1\,L_2\,M_2\sin(q_2)-2\,L_1\,(q_1)_2\,L_2\,M_2\cos(q_2)+(-4\,(q_2)_2-4\,(q_1)_2)\,(\alpha_{zz})_2+(-(q_2)_2-(q_1)_2)\,L_2^2\,M_2)/4)$

dL/dq = $\begin{pmatrix} 0 \\ -\frac{((q_1)_1\,(q_2)_1+(q_1)_1^2)\,L_1\,L_2\,M_2\sin(q_2)}{2} \end{pmatrix}$

dF/dq' = $\begin{pmatrix} (q_1)_1\,K_{11} \\ (q_2)_1\,K_{22} \end{pmatrix}$

Equazione eulero lagrange 1

$-((2\,(q_2)_1^2+4\,(q_1)_1\,(q_2)_1)\,L_1\,L_2\,M_2\sin(q_2)+(-2\,L_1\,(q_2)_2-4\,L_1\,(q_1)_2)\,L_2\,M_2\cos(q_2)-4\,(q_1)_1\,K_{11}+(-4\,(q_2)_2-4\,(q_1)_2)\,(\alpha_{zz})_2+((-(q_2)_2-(q_1)_2)\,L_2^2-4\,L_1^2\,(q_1)_2)\,M_2+(-4\,(\alpha_{zz})_1-L_1^2\,M_1)\,(q_1)_2+4\,u_1)/4=0$

Equazione eulero lagrange 2

$(2\,(q_1)_1^2\,L_1\,L_2\,M_2\sin(q_2)+2\,L_1\,(q_1)_2\,L_2\,M_2\cos(q_2)+4\,(q_2)_1\,K_{22}+(4\,(q_2)_2+4\,(q_1)_2)\,(\alpha_{zz})_2-4\,u_2+((q_2)_2+(q_1)_2)\,L_2^2\,M_2)/4=0$

(%o17) **done**

(%i18)

Robot Cilindrico

```
(%i12) cilindrico:[[q[1],L[1],0,0],[0,q[2],-%pi/2,0],[0,q[3],0,0]];
```

(%o12) $\left[[q_1, L_1, 0, 0], \left[0, q_2, -\frac{\pi}{2}, 0\right], [0, q_3, 0, 0]\right]$

```
(%i13) u:matrix([u[1]],[u[2]],[u[3]]);
```

(%o13) $\begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix}$

```
(%i14) F:matrix([K[11],0,0],[0,K[22],0],[0,0,K[33]]);
```

(%o14) $\begin{pmatrix} K_{11} & 0 & 0 \\ 0 & K_{22} & 0 \\ 0 & 0 & K_{33} \end{pmatrix}$

```
(%i15) distance:[matrix([0],[0],[-L[1]/2]),matrix([0],[-L[2]/2],[0]),matrix([0],
[0],[-L[3]/2])];
```

(%o15) $$\left[\begin{pmatrix} 0 \\ 0 \\ -\frac{L_1}{2} \end{pmatrix}, \begin{pmatrix} 0 \\ -\frac{L_2}{2} \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ -\frac{L_3}{2} \end{pmatrix}\right]$$

```
(%i16) eel(cilindrico,F,u,distance);
```

$(q_i)_i$ Indica la derivata i-esima di q_i

Energia cinetica T= $(4\,(q_1)_1^2\,(\alpha_{yy})_3 + 4\,(q_1)_1^2\,M_3\,q_3^2 - 4\,(q_1)_1^2\,L_3\,M_3\,q_3 + ((q_1)_1^2\,L_3^2 + 4\,(q_3)_1^2 + 4\,(q_2)_1^2)\,M_3 + 4\,(q_1)_1^2\,(\alpha_{yy})_2 + 4\,(q_2)_1^2\,M_2 + 4\,(q_1)_1^2\,(\alpha_{zz})_1)/8$

Energia potenziale U= $(10\,q_2 + 10\,L_1)\,M_3 + 10\,M_2\,q_2 + (5\,L_2 + 10\,L_1)\,M_2 + 5\,L_1\,M_1$

Forze di attrito F= $\frac{(q_3)_1^2\,K_{33} + (q_2)_1^2\,K_{22} + (q_1)_1^2\,K_{11}}{2}$

Forze esterne u= $\begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix}$

Lagrangiana L= $(4\,(q_1)_1^2\,(\alpha_{yy})_3 + 4\,(q_1)_1^2\,M_3\,q_3^2 - 4\,(q_1)_1^2\,L_3\,M_3\,q_3 + ((q_1)_1^2\,L_3^2 - 80\,q_2 - 80\,L_1 + 4\,(q_3)_1^2 + 4\,(q_2)_1^2)\,M_3 + 4\,(q_1)_1^2\,(\alpha_{yy})_2 - 80\,M_2\,q_2 + (-40\,L_2 - 80\,L_1 + 4\,(q_2)_1^2)\,M_2 + 4\,(q_1)_1^2\,(\alpha_{zz})_1 - 40\,L_1\,M_1)/8$

dL/dq' = $\begin{pmatrix} \frac{4\,(q_1)_1\,(\alpha_{yy})_3 + 4\,(q_1)_1\,M_3\,q_3^2 - 4\,(q_1)_1\,L_3\,M_3\,q_3 + (q_1)_1\,L_3^2\,M_3 + 4\,(q_1)_1\,(\alpha_{yy})_2 + 4\,(q_1)_1\,(\alpha_{zz})_1}{4} \\ (q_2)_1\,M_3 + (q_2)_1\,M_2 \\ (q_3)_1\,M_3 \end{pmatrix}$

d/dt dL/dq' = $((4\,(q_1)_2\,(\alpha_{yy})_3 + 4\,(q_1)_2\,M_3\,q_3^2 + (8\,(q_1)_1\,(q_3)_1 - 4\,(q_1)_2\,L_3)\,M_3\,q_3 + ((q_1)_2\,L_3^2 - 4\,(q_1)_1\,(q_3)_1\,L_3)\,M_3 + 4\,(q_1)_2\,(\alpha_{yy})_2 + 4\,(\alpha_{zz})_1\,(q_1)_2)/4;\,(q_2)_2\,M_3 + (q_2)_2\,M_2;\,(q_3)_2\,M_3)$

dL/dq = $\begin{pmatrix} 0 \\ -10\,M_3 - 10\,M_2 \\ \frac{2\,(q_1)_1^2\,M_3\,q_3 - (q_1)_1^2\,L_3\,M_3}{2} \end{pmatrix}$

dF/dq' = $\begin{pmatrix} (q_1)_1\,K_{11} \\ (q_2)_1\,K_{22} \\ (q_3)_1\,K_{33} \end{pmatrix}$

Equazione eulero lagrange 1

$(4\,(q_1)_1\,K_{11} + 4\,(q_1)_2\,(\alpha_{yy})_3 + 4\,(q_1)_2\,M_3\,q_3^2 + (8\,(q_1)_1\,(q_3)_1 - 4\,(q_1)_2\,L_3)\,M_3\,q_3 + ((q_1)_2\,L_3^2 - 4\,(q_1)_1\,(q_3)_1\,L_3)\,M_3 + 4\,(q_1)_2\,(\alpha_{yy})_2 + 4\,(\alpha_{zz})_1\,(q_1)_2 - 4\,u_1)/4{=}0$

Equazione eulero lagrange 2

$(q_2)_1\,K_{22} + ((q_2)_2 + 10)\,M_3 - u_2 + ((q_2)_2 + 10)\,M_2{=}0$

Equazione eulero lagrange 3

$\frac{2\,(q_3)_1\,K_{33} - 2\,u_3 - 2\,(q_1)_1^2\,M_3\,q_3 + ((q_1)_1^2\,L_3 + 2\,(q_3)_2)\,M_3}{2}{=}0$

(%o16) **done**

```
(%i17)
```