Continuous Acceleration and Duty Time

(HUPR)

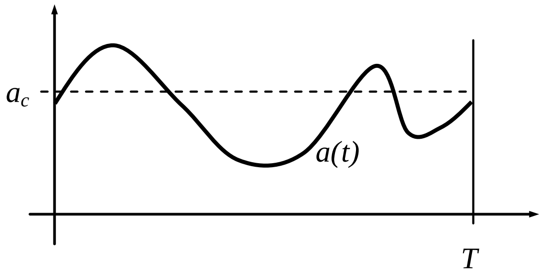
Abstract

This paper summarizes the calculation of continuous acceleration and duty time. The quantity "duty time" is introduced for an easier calculation of the continuous acceleration, which plays a major role in designing servo systems. A simple scheme for calculating the continuous acceleration based on partial duty times of a composed motion is given which is most suitable for work sheet calculations.

Continuous Acceleration

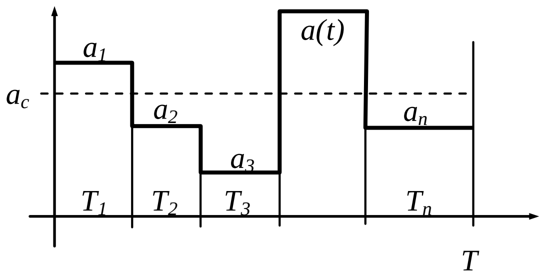
The continuous acceleration is the root mean square value of the acceleration. Given a periodic acceleration $a(t)$ with period T the continuous acceleration is defined as follows:

$$\boxed{a_c^2 T = \int_0^T a^2(t) \cdot dt}$$



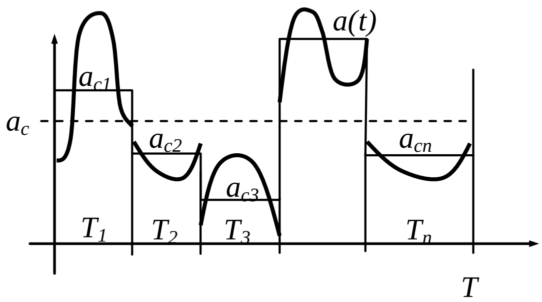
Note: For piecewise constant acceleration shapes the continuous acceleration is calculated:

$$a_c^2 T = \sum_{i=1}^{n} a_i^2 T_i$$



This relation can be generalized for general acceleration shapes, if the continuous accelerations of the pieces (a_{ci}) are known.

$$a_c^2 T = \sum_{i=1}^{n} a_{ci}^2 T_i$$



Duty Factor

The duty factor is defined as the quotient of the continuous acceleration and the (nominal) maximum acceleration:

$$\boxed{d \;:=\; \frac{a_c}{a_{\max}}}$$

Note that for composed motion cycles with given continuous accelerations a_{ci} of the partial motions one can calculate the partial duty factor

$$d_i \;:=\; \frac{a_{ci}}{a_{\max}}$$

and calculation of the (total) duty factor in terms of the partial duty factors can be done as follows:

$$d^2 T \;=\; \frac{a_c^2}{a_{\max}^2} T \;=\; \frac{\sum_{i=1}^{n} a_{ci}^2 T_i}{a_{\max}^2} \;=\; \sum_{i=1}^{n} \frac{a_{ci}^2}{a_{\max}^2} T_i \;=\; \sum_{i=1}^{n} d_i^2 T_i$$

$$\boxed{d^2 T \;=\; \sum_{i=1}^{n} d_i^2 T_i}$$

Duty Time

Now define the *duty time* as

$$\boxed{T_d \;:=\; d^2 T}$$

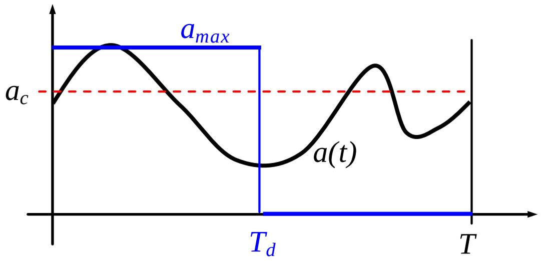
and the i-th *partial duty time* as

$$\boxed{T_{di} \;:=\; d_i^2 T_i} \quad .$$

Note, since $a^2(t)$ is proportional to the power dissipation in the motor, the dissipated energy over interval T equals

$$V = \int_0^T a^2(t) \cdot dt = a_c^2 T = a_{max}^2 d^2 T = a_{max}^2 T_d ,$$

which is the same as the dissipated energy caused by a constant continuous acceleration over interval T, and which is the same as the dissipated energy caused by a constant value of a_{max} over the *duty time* T_d .



Since

$$d^2 T = \sum_{i=1}^{n} d_i^2 T_i \qquad or \qquad T_d = \sum_{i=1}^{n} T_{di}$$

there is a very convenient scheme to calculate the continuous acceleration, which is most suitable for worksheet calculations:

Step1: Given a duty cycle with piecewise motions and maximum acceleration a_{max} calculate the partial duty time T_{di} according to

$$a_{max}^2 T_{di} = a_{ci}^2 T_i = \int_{t_i}^{t_i+T_i} a^2(t) \cdot dt .$$

In general this calculation is done by the MATLAB function DUTY, which returns the duty time of a motion profile. Note, however, that a_{max} is a fixed nominal value for the maximum acceleration (not always identical with the actual maximum acceleration of a piece of motion)!

Step 2: Calculate the (total) duty time by adding up the partial duty times:

$$T_d = \sum_{i=1}^{n} T_{di}$$

Step 3: Given the duty time T_d and period T calculate the duty factor:

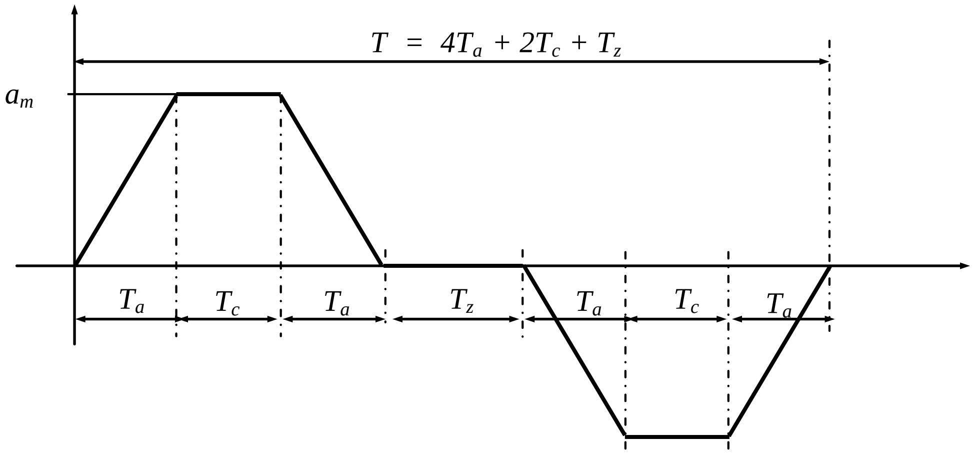
$$d = \sqrt{\frac{T_d}{T}}$$

Step 4: Calculate the continuous acceleration

$$a_c = d \cdot a_{max}$$

Duty Time Calculation for Trapezoidal Acceleration

Given a trapezoidal acceleration profile



the duty time is calculated by

$$T_d = 4T_{da} + 2T_{dc} + T_{dz}$$

where

$$T_{da} = \frac{1}{a_{\max}^2}\int_0^{T_a} a^2(t)\cdot dt = \frac{1}{a_{\max}^2}\int_0^{T_a} a_m^2 \frac{t^2}{T_a^2}\cdot dt = \frac{1}{3}\frac{a_m^2}{a_{\max}^2} T_a$$

$$T_{dc} = \frac{a_m^2}{a_{\max}^2} T_c$$

$$T_{dz} = 0 \ .$$

Thus the duty time of a trapezoidal acceleration profile equals

$$T_d = (4/3\ T_a + 2T_c)\ \frac{a_m^2}{a_{\max}^2}.$$

Note that the quotient $\frac{a_m^2}{a_{\max}^2}$ only equals 1 if the actual maximum acceleration a_m is equal to the nominal maximum acceleration a_{max}.

Appendix A - Matlab function DUTY and EXCEL Makro

```
function [td,acont,du] = duty(smax,vmax,amax,stime,unit,infotext)
%
% DUTY calculate duty time, continuous acceleration and duty cycle of a
%      motion profile
%
%      [tduty,ac,du] = duty(smax,vmax,amax,stime,unit) % return cont. acceleration
%      duty(smax,vmax,amax,stime,unit)                % plot acc. profile
%
%      Theory:
%          Assume an acceleration profile with the following phases
%          1) ramp up from    0 .. am  over time Ta
%          2) constant phase am        over time Tc
%          3) ramp down from  am .. 0  over time Ta
%          4) beeing zero              over time Tz
%          5) ramp down from  0 .. -am over time Ta
%          6) constant phase  -am      over time Tc
%          7) ramp up from    -am .. 0 over time Ta
%
%        Note that the total time equals T = 4*Ta + 2*Tc + Tz !
%
%        Then:
%                  Duty time is tduty = (4/3*Ta + 2*Tc)*(am/amax)^2
%            continuous acceleration equals   ac = am * sqrt([4/3*Ta + 2*Tc]/T)
%        Duty factor is defined by  du = ac/am
%
%        See also: MOTION
%

   if (nargin < 2) vmax = 1000; end
   if (nargin < 3) amax = 10000; end
   if (nargin < 4) stime = 0; end
   if (nargin < 5) unit = 'mm'; end
   if (nargin < 6) infotext = 'acceleration profile'; end

   [T,tsva] = motion(smax,vmax,amax,stime,unit);

   t = tsva(:,1);    a = tsva(:,4);

   % continuous acceleration must be based on amax, not on the maximum
   % acceleration reached during the movement! (MILO 30.1.02)

   am = max(abs(a));

   ta = t(2) - t(1);
   tc = t(3) - t(2);
   tz = t(5) - t(4);

   tduty = [4/3*ta + 2*tc]*(am/amax)^2;
   d = sqrt(tduty/T);    % duty
   ac = amax * d;

   if (nargout == 0)
```

```
        hold off
        plot(t*1000,a,'r');
        hold on
        plot(get(gca,'xlim'),[0 0],'k');
        plot([0 T*1000],[ac ac],'r-.');
        title(sprintf(['duty  time  %g,  max  acc.  %g,  cont.  acc.  %g,  duty  %g
%%'],rd(tduty*1000),rd(am),rd(ac),rd(d*100)));
        ylabel(sprintf(['smax   =   %g,   vmax   =   %g,   amax   =   %g,   jtime   =
%g'],max(smax),vmax,amax,stime));
        xlabel(sprintf('T  =  %g  ms  (jerk:  %g  ms,  const.  acc.  %g  ms,  zero  acc  %g
ms)',rd(T*1000),rd(ta*1000),rd(tc*1000),rd(tz*1000)));
        shg
    else
        td = tduty;
        du = d;
        acont = ac;
    end
    ac = 0;
    return

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
% auxillary functions

function y = rd(x)    % round to one digit after comma
    y = round(10*x)/10;
    return

% eof
```

EXCEL Makro

```
Dim Matlab As Object

Function Matlab_Duty(Dist, vel, acc, sTime)   ' calculate duty time
    Dim Command As String
    If Matlab Is Nothing Then
        Start
    End If
    Command = " a=duty(" + Str$(Dist) + "," + Str$(vel) + "," + Str$(acc)
    Command = Command + "," +    Str$(sTime) + ")"
    Resultstr = Matlab.Execute(Command)
    Resultstr = Right(Resultstr, Len(Resultstr) - 7)
    Resultstr = Left(Resultstr, Len(Resultstr) - 2)
    Matlab_Duty = Val(Resultstr)
    State = 2
End Function

Function Start()
    Dim Resultstr As String
    Set Matlab = CreateObject("MatLab.Application")
    Start = 1
End Function
```

Example:

```
» motion(120,2000,40000,0.03)
» duty(120,2000,40000,0.03)
```