new_tip_dist-1

April 27, 2022

[1]:
```
import numpy as np
from matplotlib import pyplot as plt

from IPython.display import display, Latex
```

[2]:
```
# sigularities
def sing(x,a,n):
    if not isinstance(x, np.ndarray):
        x = np.array([x])
    ni = np.zeros(x.size)
    for i in range(x.size):
        if x[i] >=a and n>=0:
            ni[i] = (x[i]-a)**n
    return ni/np.math.factorial(n)
```

[3]:
```
def lx(ord_f,pc, tire_l, case,ab,p1,p2):
    react_f = GVWR*((1-pc)*sing(dx,0,ord_f)+pc/2*(sing(dx,lw1,ord_f)+sing(dx,␣
↪lw2, ord_f)))
    other_f = p1*sing(dx,a2,ord_f)+p2*sing(dx,lc,ord_f)+W_tip/
↪base_len*(sing(dx,0, ord_f+1)-sing(dx,base_len, ord_f+1))
    if case == 1:
        return react_f - other_f - Rc*sing(dx, tire_l, ord_f)
    else:
        return react_f - other_f
```

[4]:
```
def pp(tire_l,load_fac,ab):
    react_f = Rc*((tire_l-ab)*load_fac+wheel_base)+W_tip*(G_tip-ab)
    p2 = react_f/(lc-ab)
    p1 = load_fac * Rc + W_tip - p2
    # print(f'p1<0: {ab==a1}, p1:{p1}')
    return p1,p2
```

[5]:
```
def trail(ord_f,pc,tire_len, case,ab,p1,p2):
    react_f = p1*sing(dx,a2,ord_f)+p2*sing(dx,lc,ord_f)-W_tip/
↪tip_len*(sing(dx,a1, ord_f+1)-sing(dx,a1+tip_len, ord_f+1))#todo␣
↪sigularity(a), mom = @sig(a,1,2)
    if case == 1:
```

```
        rc_f = Rc*sing(dx,tire_len,ord_f)
    else:
        rc_f = Rc*(sing(dx,tire_len+wheel_base,ord_f)+sing(dx,tire_len,ord_f))
    return react_f - rc_f
```

[6]:
```
def plot_trail(pl,l):

    lims = pl.axis()
    dy = lims[3]-lims[2]
    dx = lims[1]-lims[0]
    sca = dy/dx

    points = np.array([[0, base_len,␣
↪base_len,base_len-6,base_len-12,base_len-36,base_len-48,a0,a0,0,0],
                    [0,0,8,8,12,12,8,8,6,6,0]])

    points_2 = np.array([[-6,6,6,-6,-6],
                     [0,0,2,2,0]])

    points_2[0] += a2

    dash_p = np.array([[a0,a0],[0,8]])
    tip_p = np.array([[a1,lf,lf-6,a1,a1],
                [8,8,2,2,8]])

    for i in [points,points_2,dash_p,tip_p]:
         i[1] = sca*i[1]

    pl.plot(*points,'c', label='Base (Reference)')
    pl.plot(*points_2,'c')
    pl.plot(*dash_p, 'c--')
    pl.plot(*tip_p,'y', label='Tip Deck (Refernce)')
    plot_wheels(pl,sca,l)
```

[7]:
```
len_full = 300

# predefined arrays
dx = np.linspace(-1,len_full, 200)
cat_tire_l_arr = np.arange(60,len_full-50,10)
```

[8]:
```
def plot_wheels(pl,sca,l):
    wheel_loc = [lw1,lw2]
    wheel_loc2 = [l,l+48]

    r = np.array([12,8])

    wheels = []
```

```
    t = np.linspace(0, 2*np.pi, 100)
    for i in wheel_loc:
        pl.plot(i+r[0]*np.cos(t),(-0.5+np.sin(t))*r[0]*sca,'c')

    pl.plot(wheel_loc2[0]+r[1]*np.cos(t),(8+r[1]+r[1]*np.sin(t))*sca,'m',␣
↪label='Bobcat Wheels (load locations)')
    pl.plot(wheel_loc2[1]+r[1]*np.cos(t),(8+r[1]+r[1]*np.sin(t))*sca,'m')
```

[9]:
```
def plot_x(react_mat,react_mattip,max_p, f_arr):
    # SFD BMD, \u03C3 vs distance for each condition of len,percent

    lft = [react_mat,react_mattip]
    plt_n = ['main', 'tip']
    for ii in range(len(react_mat)):
        fig, ax = plt.subplots(1,2)
        for i in range(2):
            ax[i].grid(True)
            m_half = lft[i][ii]

            ax[i].plot(dx,m_half[0], 'b',label='Shear (lb)')
            ax[i].plot(dx,m_half[1]*1e-2,'r',label='Moment(100*lb*in)')
            ax[i].plot(dx,m_half[2]*1e-1, 'g',label='Sigma (10*psi)')


            plot_trail(ax[i],f_arr[ii])
            ax[i].legend()
            ax[i].set_title(f'SFD BMD, \u03C3 allong trailer(in) for current␣
↪loading on {plt_n[i]}')
        fig.suptitle(f'Plots for L= {round(f_arr[ii],2)}(in) rear Load:
↪{int(max_p[ii,0]*100)}%')
```

[10]:
```
def run_f(perc=None):
    max_p = [[0,0]]
    react_mat = []
    display(Latex(f'Conststants: \n$R={GVWR}\na_1={a1}, a_2={a2}\nL_c={lc},␣
↪L_{{w,1}}= {lw1}, L_{{w,2}} = {lw2}, $'))

    max_ptip = [[0,0]]
    react_mattip = []
    # loop through locations
    #const
    const_v = Rc*wheel_base+W_tip*(G_tip+G_base)
    const_react = GVWR*(lw1+lw2)/2
    if perc:
        f_arr = [perc]
    else:
        f_arr = cat_tire_l_arr
```

```
    for par_v in f_arr:
        if perc:
            p = par_v
            cat_tire_l = (const_react*p-const_v)/(2*Rc)
        else:
            cat_tire_l = par_v
            p = (2*Rc*cat_tire_l+const_v)/const_react
        ab = a2
        if cat_tire_l < a1:
            case = 1

        else:
            case = 2
        if cat_tire_l > lc+(lc-G_tip)*W_tip/(Rc*case)-48/case:
            ab = a1

        # initialize constants for each

        p1,p2 = pp(cat_tire_l,case,ab)
        r1 = round((1-p)*GVWR,1)
        r2 = round(p*GVWR,1)

        #singularity
        display(Latex(f'$\\text{{Equation solution: }} R_1={r1},R_2={r2}, p_1={round(p1,1)}, p_2={round(p2,1)}$'))
        load = lx(0,p,cat_tire_l,case,ab,p1,p2)
        mom = lx(1,p,cat_tire_l,case,ab, p1,p2)

        tip_l = trail(0,p,cat_tire_l,case,ab,p1,p2)
        tip_m = trail(1,p, cat_tire_l,case,ab,p1,p2)
        #, v: {}, m:{}')
        sig = mom/(rail_cnt*shear_mod)  # stress
        sigtip = tip_m/(rail_cnt2*shear_mod)

        react_mat.append([load, mom, sig])
        max_sig = np.max(np.abs(sig))

        max_p.append([p*1, max_sig*1])  # max stress for this loading condition and this location

        react_mattip.append([tip_l, tip_m, sigtip])
        max_sigtip = np.max(np.abs(sigtip))

        max_ptip.append([p*1, max_sigtip*1])  # max stress for this loading condition and this location

    # tabulation of this location, and max of location
```

```
    max_p = np.array(max_p)
    m_n = np.argmax(max_p,0)
    m_a = max_p[m_n[1],:]

    # adding to list of all locs

    # tabulation of this location, and max of location
    max_ptip = np.array(max_ptip)
    m_ntip = np.argmax(max_ptip,0)
    m_atip = max_ptip[m_ntip[1],:]
    print_conditions(max_p,f_arr,m_n,m_a,max_ptip,m_ntip,m_atip)

    return react_mat,react_mattip,max_p, f_arr
```

[11]:
```
def print_conditions(max_p,f_arr,m_n,m_a,max_ptip,m_ntip,m_atip):
    # print(full r)
    def prt_1(ma):
        for i in range(ma.shape[0]-1):
            fis = ma[i+1,1]
            fs2 = '\dashv \\vdash' if fis>= yield_s else ''
            # st3 = '\Rightarrow'
            display(Latex(f'$L= {round(f_arr[i],1)}(in),␣
↪x_{{wheel}}={int(ma[i+1,0]*100)} \%, \sigma_{{max}}={round(fis,2)}(psi)␣
↪\Rightarrow {round(fis/1000,1)}(ksi){fs2}$'))

    def prt_2(mn,ma):
        print(f'\n----------')
        display(Latex(f'$\sigma_{{Max,Overall}}:: L={round(f_arr[mn[1]-1],␣
↪2)}(in), R_2={int(ma[0]*100)}%,  \sigma = {round(ma[1], 2)}(psi)$'))

    display(Latex(f'NOTE: $\dashv \\vdash \\text{{means failure at }}␣
↪\sigma_y={round(yield_s/1000,1)}$'))

    # max for each percent, len

    prt_1(max_p)
    prt_2(m_n,m_a)
    # max for each percent, len
    print(f'\n\n--------------------\ntrailer\n--------------------\n')

    prt_1(max_ptip)

    prt_2(m_ntip,m_atip)
```

[12]:
```
#test 1
#constants
yield_s = 50000
```

```
shear_mod = 5.61
shear_mod_tip=5.49

rail_cnt = 2

rail_cnt2 = 2

GVWR = 16000

W_tip = 1800 # tare/2
Rc = GVWR/2 - W_tip

lc = 200-12
a1 = 104
a0 = 56

tip_len = 194
base_len = 220

G_base = base_len / 2
G_tip = tip_len / 2 + a1

a2 = a1 + 16
lw1 = lc -21
lw2 = lc +15
lf = a1+tip_len


wheel_base = 48

react_mat1,react_mattip1,mp,fa = run_f()
plot_x(react_mat1,react_mattip1,mp,fa)
```

[12]: Conststants: $R = 16000 a_1 = 104, a_2 = 120 L_c = 188, L_{w,1} = 167, L_{w,2} = 203,$

[12]: Equation solution: $R_1 = 7343.8, R_2 = 8656.2, p_1 = 6950.0, p_2 = 1050.0$

[12]: Equation solution: $R_1 = 6673.5, R_2 = 9326.5, p_1 = 6038.2, p_2 = 1961.8$

[12]: Equation solution: $R_1 = 6003.2, R_2 = 9996.8, p_1 = 5126.5, p_2 = 2873.5$

[12]: Equation solution: $R_1 = 5333.0, R_2 = 10667.0, p_1 = 4214.7, p_2 = 3785.3$

[12]: Equation solution: $R_1 = 4662.7, R_2 = 11337.3, p_1 = 3302.9, p_2 = 4697.1$

[12]: Equation solution: $R_1 = 3992.4, R_2 = 12007.6, p_1 = 9502.9, p_2 = 4697.1$

[12]: Equation solution: $R_1 = 3322.2, R_2 = 12677.8, p_1 = 7679.4, p_2 = 6520.6$

[12]: Equation solution: $R_1 = 2651.9, R_2 = 13348.1, p_1 = 5855.9, p_2 = 8344.1$

[12]: Equation solution: $R_1 = 1981.6, R_2 = 14018.4, p_1 = 4032.4, p_2 = 10167.6$

[12] : Equation solution: $R_1 = 1311.4, R_2 = 14688.6, p_1 = 2208.8, p_2 = 11991.2$

[12] : Equation solution: $R_1 = 641.1, R_2 = 15358.9, p_1 = 385.3, p_2 = 13814.7$

[12] : Equation solution: $R_1 = -29.2, R_2 = 16029.2, p_1 = -1164.3, p_2 = 15364.3$

[12] : Equation solution: $R_1 = -699.5, R_2 = 16699.5, p_1 = -2640.5, p_2 = 16840.5$

[12] : Equation solution: $R_1 = -1369.7, R_2 = 17369.7, p_1 = -4116.7, p_2 = 18316.7$

[12] : Equation solution: $R_1 = -2040.0, R_2 = 18040.0, p_1 = -5592.9, p_2 = 19792.9$

[12] : Equation solution: $R_1 = -2710.3, R_2 = 18710.3, p_1 = -7069.0, p_2 = 21269.0$

[12] : Equation solution: $R_1 = -3380.5, R_2 = 19380.5, p_1 = -8545.2, p_2 = 22745.2$

[12] : Equation solution: $R_1 = -4050.8, R_2 = 20050.8, p_1 = -10021.4, p_2 = 24221.4$

[12] : Equation solution: $R_1 = -4721.1, R_2 = 20721.1, p_1 = -11497.6, p_2 = 25697.6$

[12] : NOTE: $\dashv\vdash$ means failure at $\sigma_y = 50.0$

[12] : $L = 60(in), x_{wheel} = 54\%, \sigma_{max} = 40134.66(psi) \Rightarrow 40.1(ksi)$

[12] : $L = 70(in), x_{wheel} = 58\%, \sigma_{max} = 39847.73(psi) \Rightarrow 39.8(ksi)$

[12] : $L = 80(in), x_{wheel} = 62\%, \sigma_{max} = 40418.74(psi) \Rightarrow 40.4(ksi)$

[12] : $L = 90(in), x_{wheel} = 66\%, \sigma_{max} = 39723.63(psi) \Rightarrow 39.7(ksi)$

[12] : $L = 100(in), x_{wheel} = 70\%, \sigma_{max} = 37839.2(psi) \Rightarrow 37.8(ksi)$

[12] : $L = 110(in), x_{wheel} = 75\%, \sigma_{max} = 37446.54(psi) \Rightarrow 37.4(ksi)$

[12] : $L = 120(in), x_{wheel} = 79\%, \sigma_{max} = 30278.39(psi) \Rightarrow 30.3(ksi)$

[12] : $L = 130(in), x_{wheel} = 83\%, \sigma_{max} = 23110.24(psi) \Rightarrow 23.1(ksi)$

[12] : $L = 140(in), x_{wheel} = 87\%, \sigma_{max} = 15942.09(psi) \Rightarrow 15.9(ksi)$

[12] : $L = 150(in), x_{wheel} = 91\%, \sigma_{max} = 9400.85(psi) \Rightarrow 9.4(ksi)$

[12] : $L = 160(in), x_{wheel} = 95\%, \sigma_{max} = 9846.79(psi) \Rightarrow 9.8(ksi)$

[12] : $L = 170(in), x_{wheel} = 100\%, \sigma_{max} = 8632.43(psi) \Rightarrow 8.6(ksi)$

[12] : $L = 180(in), x_{wheel} = 104\%, \sigma_{max} = 12730.78(psi) \Rightarrow 12.7(ksi)$

[12] : $L = 190(in), x_{wheel} = 108\%, \sigma_{max} = 19899.09(psi) \Rightarrow 19.9(ksi)$

[12] : $L = 200(in), x_{wheel} = 112\%, \sigma_{max} = 27067.39(psi) \Rightarrow 27.1(ksi)$

[12] : $L = 210(in), x_{wheel} = 116\%, \sigma_{max} = 34235.7(psi) \Rightarrow 34.2(ksi)$

[12] : $L = 220(in), x_{wheel} = 121\%, \sigma_{max} = 41404.01(psi) \Rightarrow 41.4(ksi)$

[12] : $L = 230(in), x_{wheel} = 125\%, \sigma_{max} = 48572.31(psi) \Rightarrow 48.6(ksi)$

[12] : $L = 240(in), x_{wheel} = 129\%, \sigma_{max} = 55740.62(psi) \Rightarrow 55.7(ksi)$ $\dashv\vdash$

----

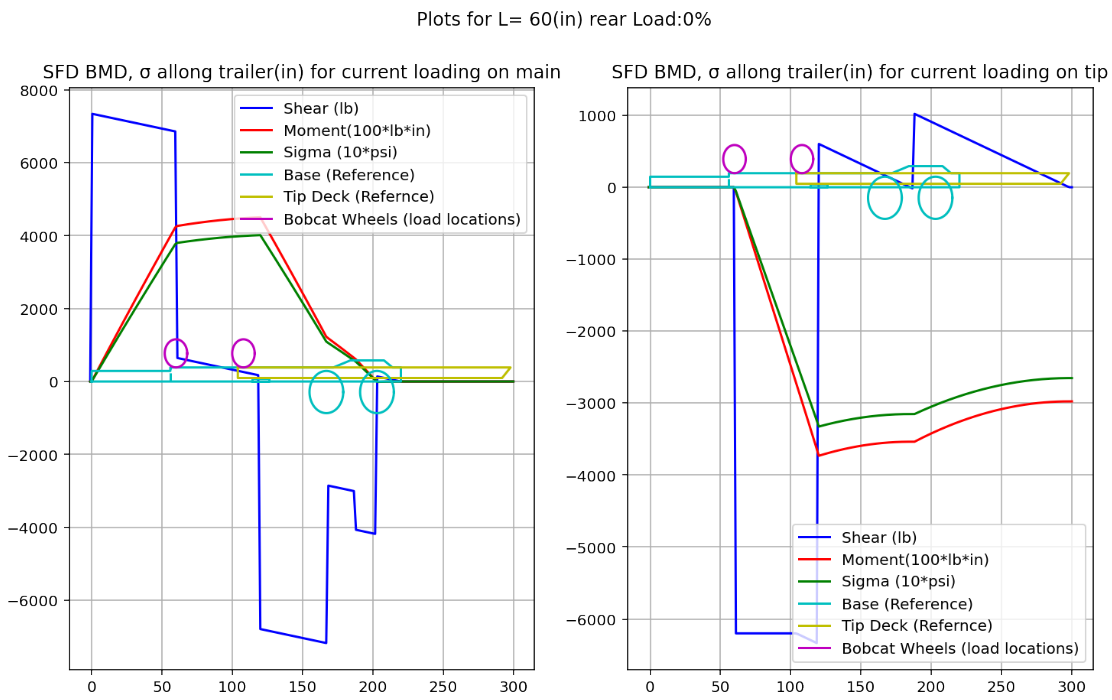
[12]: $\sigma_{Max,Overall} :: L = 240(in), R_2 = 129$

```
--------------------
trailer
--------------------
```

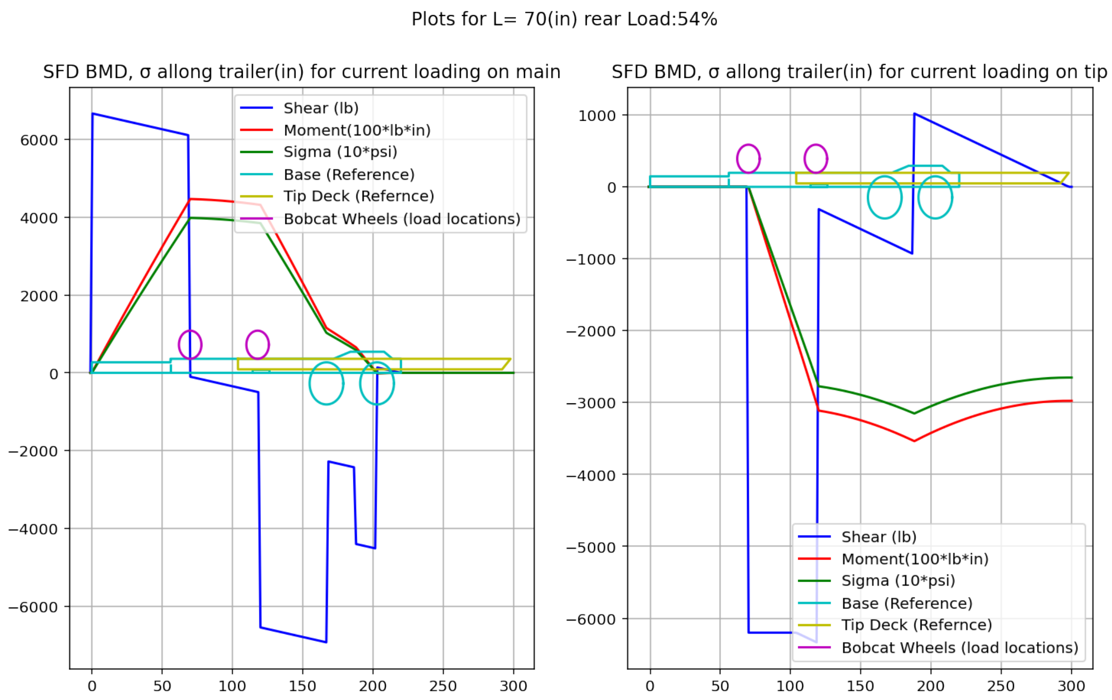
[12]: $L = 60(in), x_{wheel} = 54\%, \sigma_{max} = 33260.66(psi) \Rightarrow 33.3(ksi)$
[12]: $L = 70(in), x_{wheel} = 58\%, \sigma_{max} = 31520.7(psi) \Rightarrow 31.5(ksi)$
[12]: $L = 80(in), x_{wheel} = 62\%, \sigma_{max} = 31520.7(psi) \Rightarrow 31.5(ksi)$
[12]: $L = 90(in), x_{wheel} = 66\%, \sigma_{max} = 31520.7(psi) \Rightarrow 31.5(ksi)$
[12]: $L = 100(in), x_{wheel} = 70\%, \sigma_{max} = 31520.7(psi) \Rightarrow 31.5(ksi)$
[12]: $L = 110(in), x_{wheel} = 75\%, \sigma_{max} = 5630.28(psi) \Rightarrow 5.6(ksi)$
[12]: $L = 120(in), x_{wheel} = 79\%, \sigma_{max} = 4996.63(psi) \Rightarrow 5.0(ksi)$
[12]: $L = 130(in), x_{wheel} = 83\%, \sigma_{max} = 4996.63(psi) \Rightarrow 5.0(ksi)$
[12]: $L = 140(in), x_{wheel} = 87\%, \sigma_{max} = 6542.57(psi) \Rightarrow 6.5(ksi)$
[12]: $L = 150(in), x_{wheel} = 91\%, \sigma_{max} = 10483.61(psi) \Rightarrow 10.5(ksi)$
[12]: $L = 160(in), x_{wheel} = 95\%, \sigma_{max} = 16009.45(psi) \Rightarrow 16.0(ksi)$
[12]: $L = 170(in), x_{wheel} = 100\%, \sigma_{max} = 19875.0(psi) \Rightarrow 19.9(ksi)$
[12]: $L = 180(in), x_{wheel} = 104\%, \sigma_{max} = 23295.76(psi) \Rightarrow 23.3(ksi)$
[12]: $L = 190(in), x_{wheel} = 108\%, \sigma_{max} = 27782.82(psi) \Rightarrow 27.8(ksi)$
[12]: $L = 200(in), x_{wheel} = 112\%, \sigma_{max} = 36729.43(psi) \Rightarrow 36.7(ksi)$
[12]: $L = 210(in), x_{wheel} = 116\%, \sigma_{max} = 45676.04(psi) \Rightarrow 45.7(ksi)$
[12]: $L = 220(in), x_{wheel} = 121\%, \sigma_{max} = 54622.64(psi) \Rightarrow 54.6(ksi) \dashv\vdash$
[12]: $L = 230(in), x_{wheel} = 125\%, \sigma_{max} = 63569.25(psi) \Rightarrow 63.6(ksi) \dashv\vdash$
[12]: $L = 240(in), x_{wheel} = 129\%, \sigma_{max} = 72515.86(psi) \Rightarrow 72.5(ksi) \dashv\vdash$

```
----------
```

[12]: $\sigma_{Max,Overall} :: L = 240(in), R_2 = 129$
[12]:

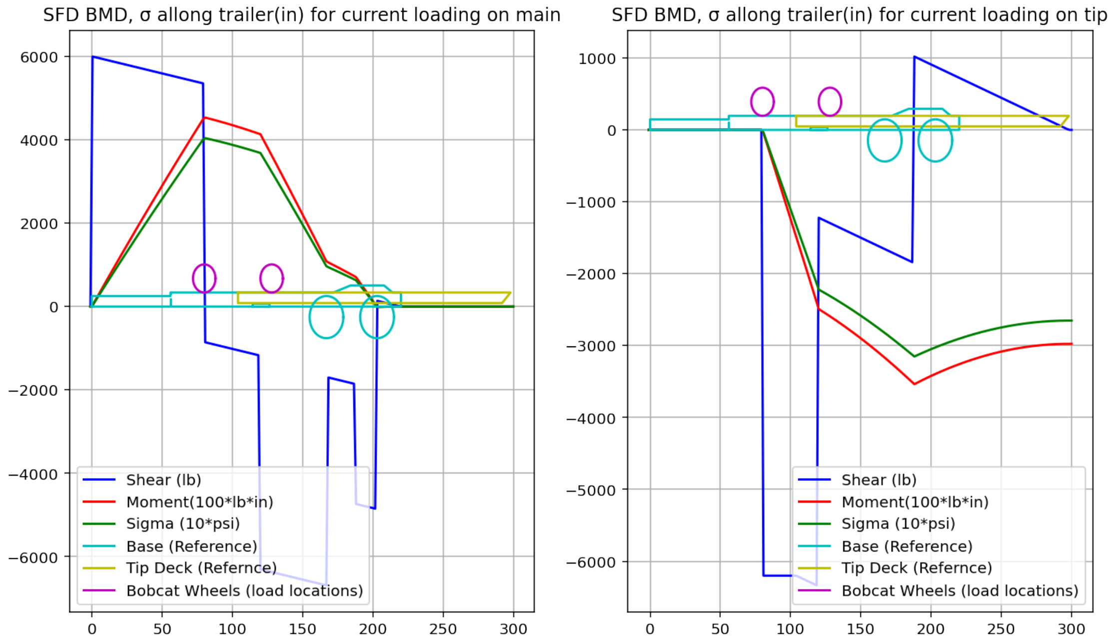

Plots for L= 60(in) rear Load:0%
SFD BMD, σ allong trailer(in) for current loading on main
SFD BMD, σ allong trailer(in) for current loading on tip
Shear (lb)
Moment(100*lb*in)
Sigma (10*psi)
Base (Reference)
Tip Deck (Refernce)
Bobcat Wheels (load locations)

[12]:


Plots for L= 70(in) rear Load:54%
SFD BMD, σ allong trailer(in) for current loading on main
SFD BMD, σ allong trailer(in) for current loading on tip
Shear (lb)
Moment(100*lb*in)
Sigma (10*psi)
Base (Reference)
Tip Deck (Refernce)
Bobcat Wheels (load locations)

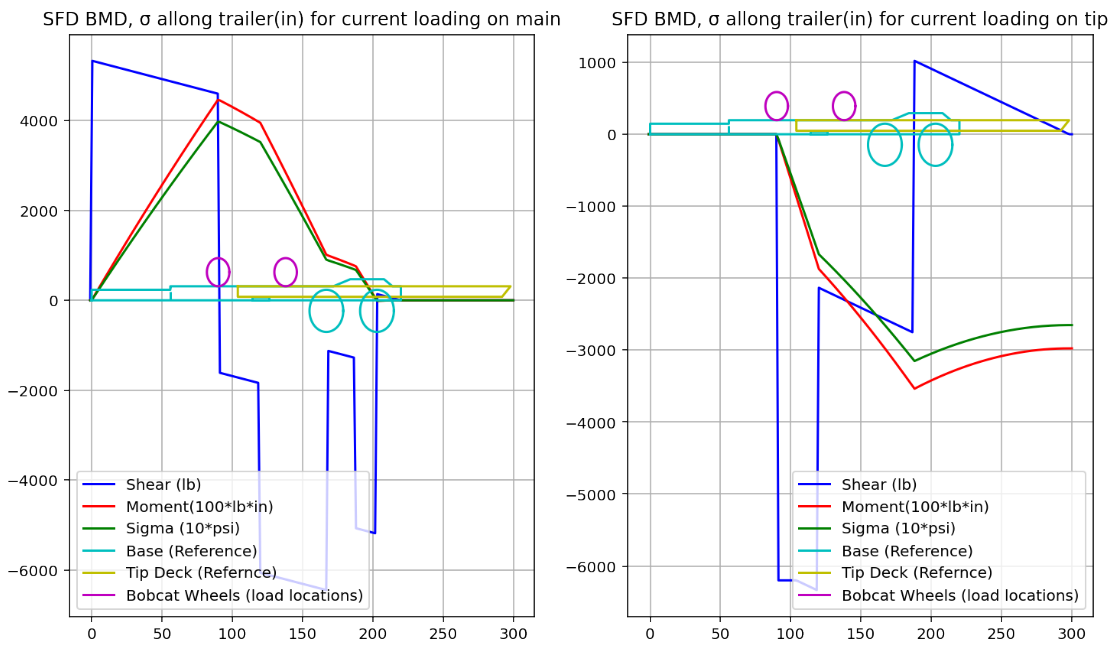
[12]:

Plots for L= 80(in) rear Load:58%

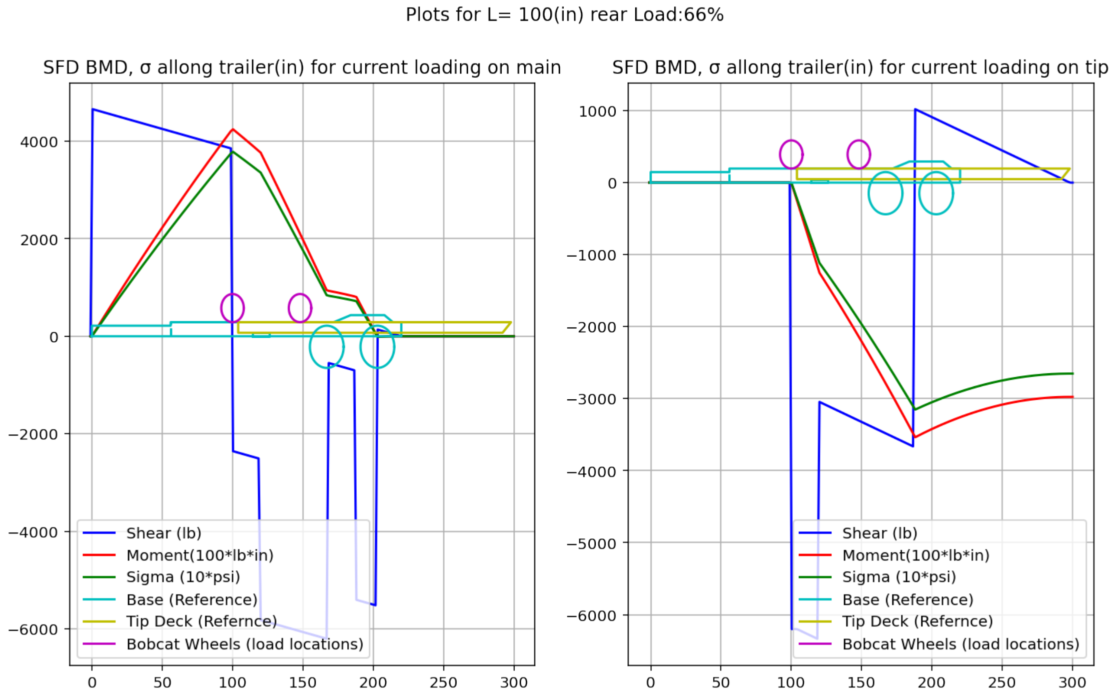


[12]:

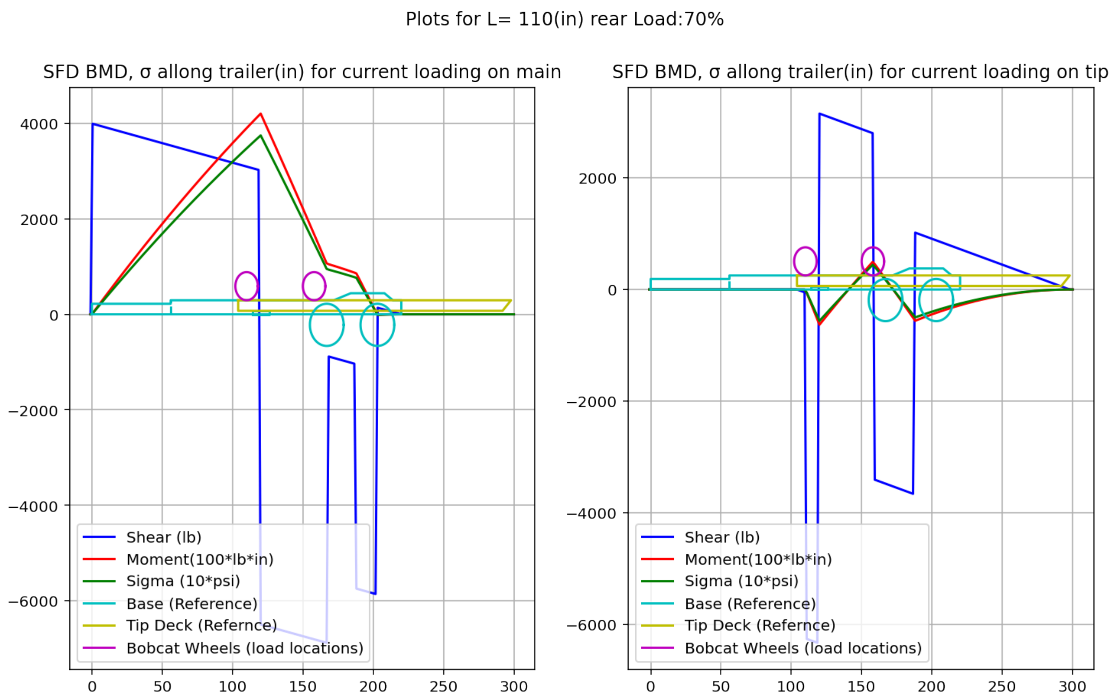
Plots for L= 90(in) rear Load:62%



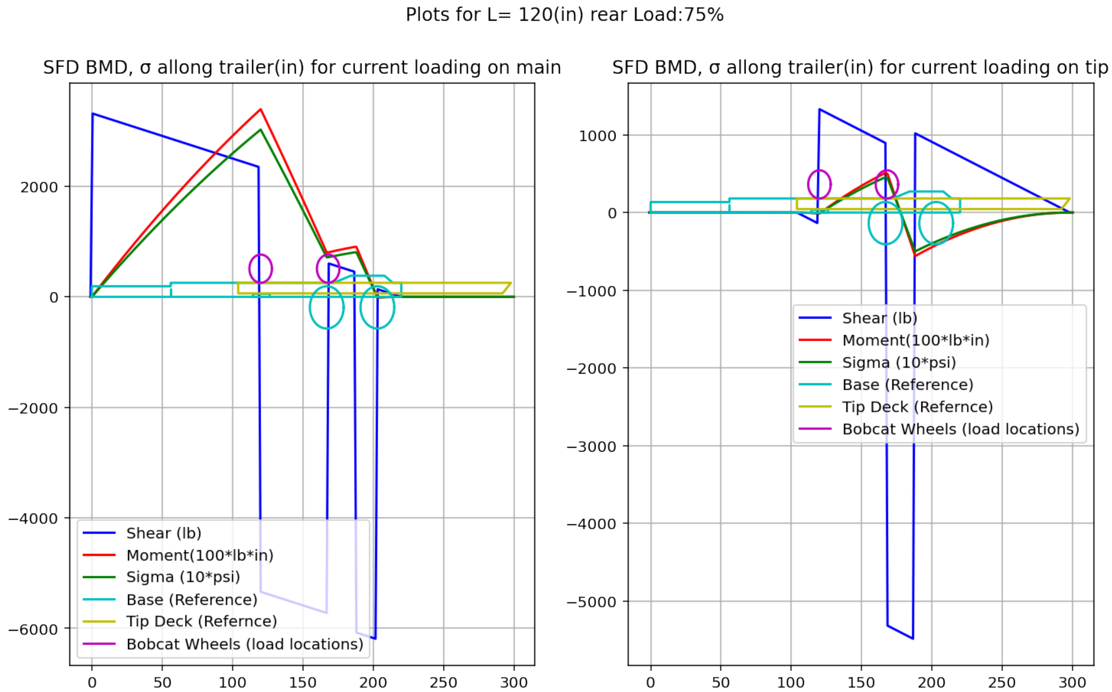
[12]:


Plots for L= 100(in) rear Load:66%
SFD BMD, σ allong trailer(in) for current loading on main
SFD BMD, σ allong trailer(in) for current loading on tip
Shear (lb)
Moment(100*lb*in)
Sigma (10*psi)
Base (Reference)
Tip Deck (Refernce)
Bobcat Wheels (load locations)

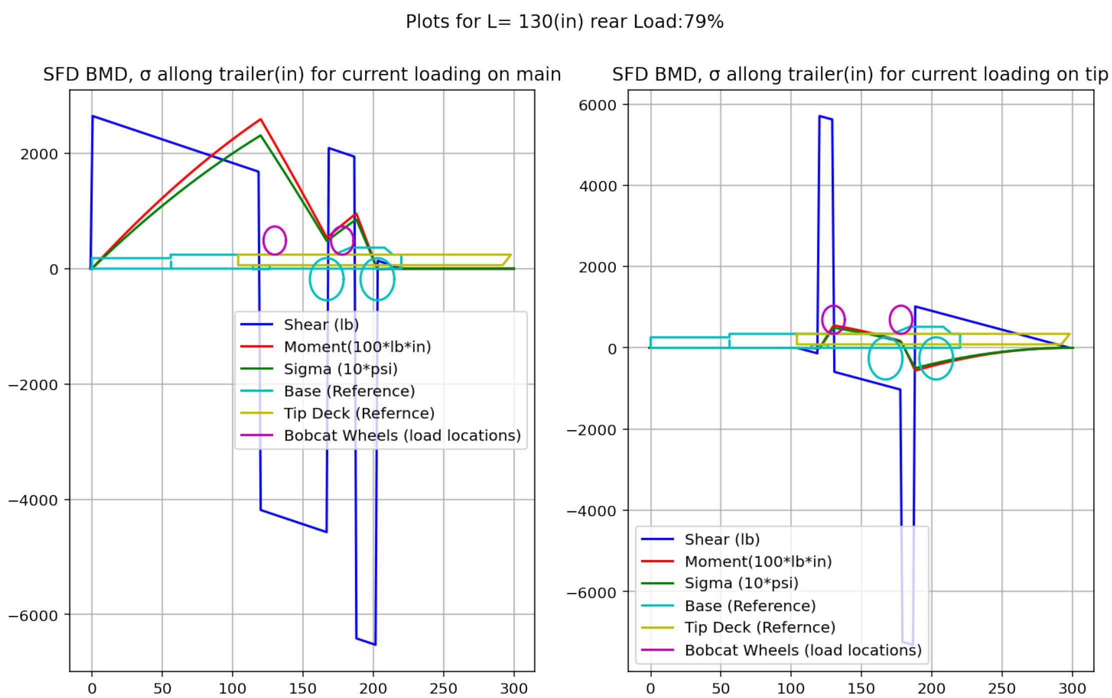
[12]:


Plots for L= 110(in) rear Load:70%
SFD BMD, σ allong trailer(in) for current loading on main
SFD BMD, σ allong trailer(in) for current loading on tip
Shear (lb)
Moment(100*lb*in)
Sigma (10*psi)
Base (Reference)
Tip Deck (Refernce)
Bobcat Wheels (load locations)

[12]:

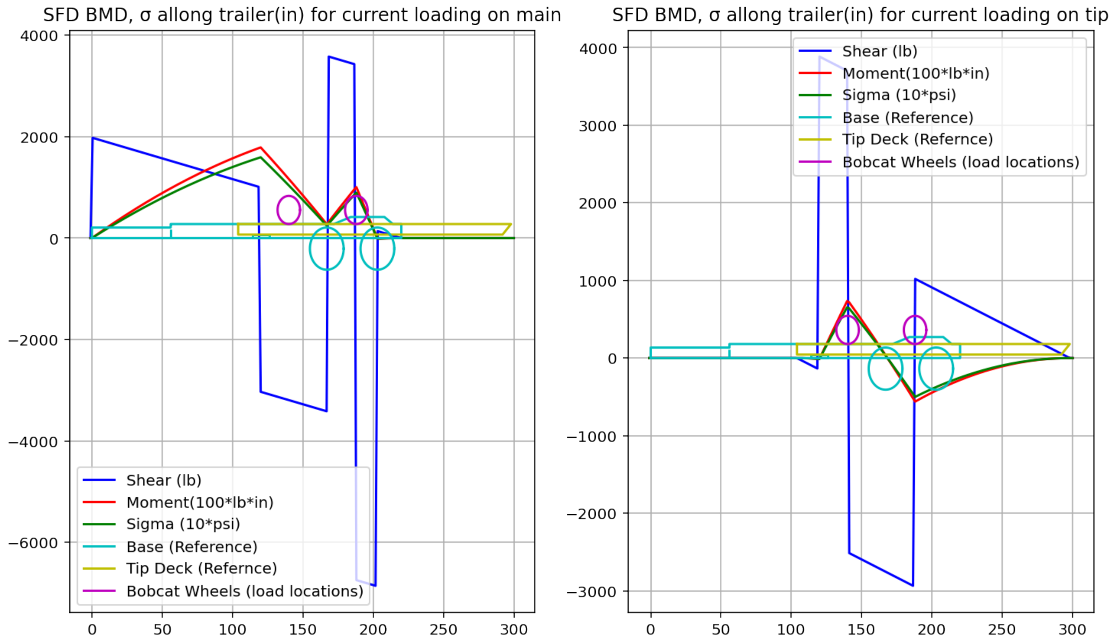

Plots for L= 120(in) rear Load:75%
SFD BMD, σ allong trailer(in) for current loading on main
SFD BMD, σ allong trailer(in) for current loading on tip
Shear (lb)
Moment(100*lb*in)
Sigma (10*psi)
Base (Reference)
Tip Deck (Refernce)
Bobcat Wheels (load locations)

[12]:


Plots for L= 130(in) rear Load:79%
SFD BMD, σ allong trailer(in) for current loading on main
SFD BMD, σ allong trailer(in) for current loading on tip
Shear (lb)
Moment(100*lb*in)
Sigma (10*psi)
Base (Reference)
Tip Deck (Refernce)
Bobcat Wheels (load locations)

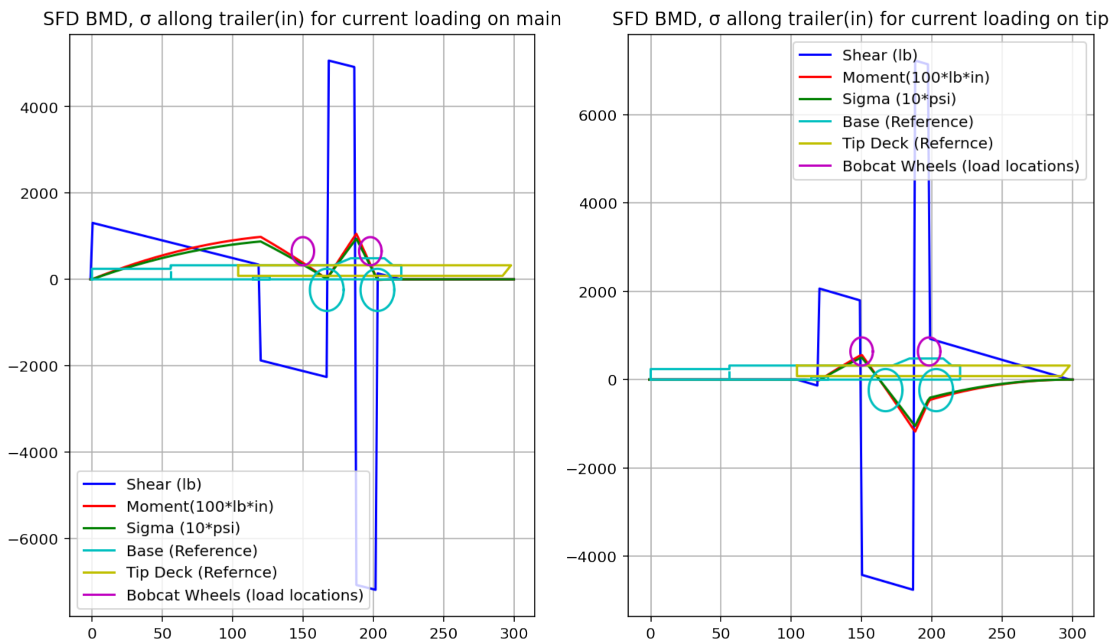
[12]:

Plots for L= 140(in) rear Load:83%



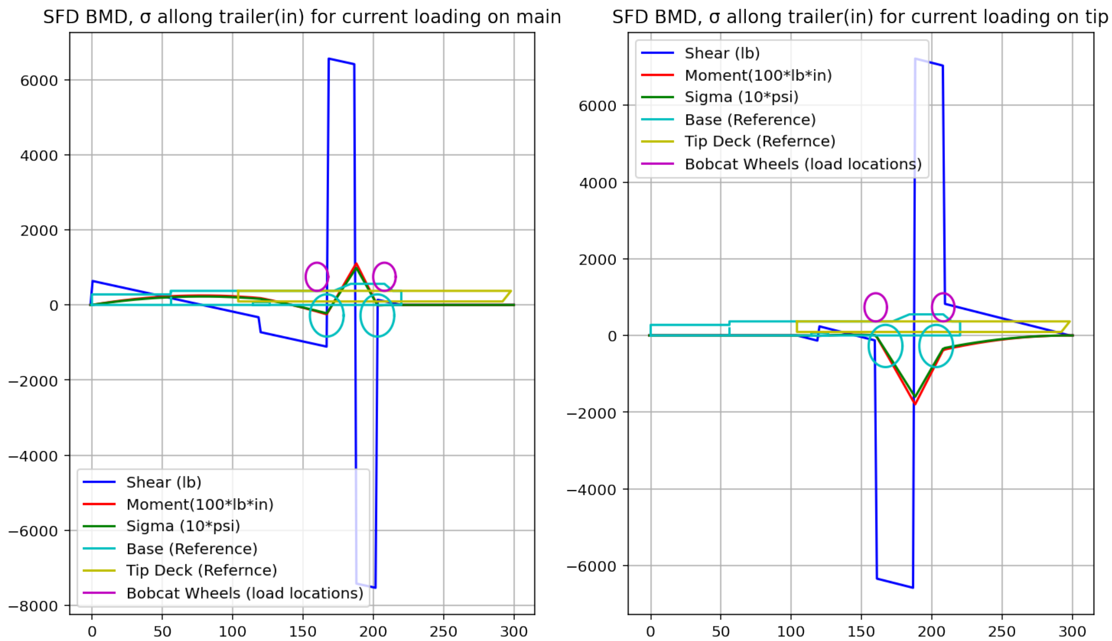
[12]:

Plots for L= 150(in) rear Load:87%



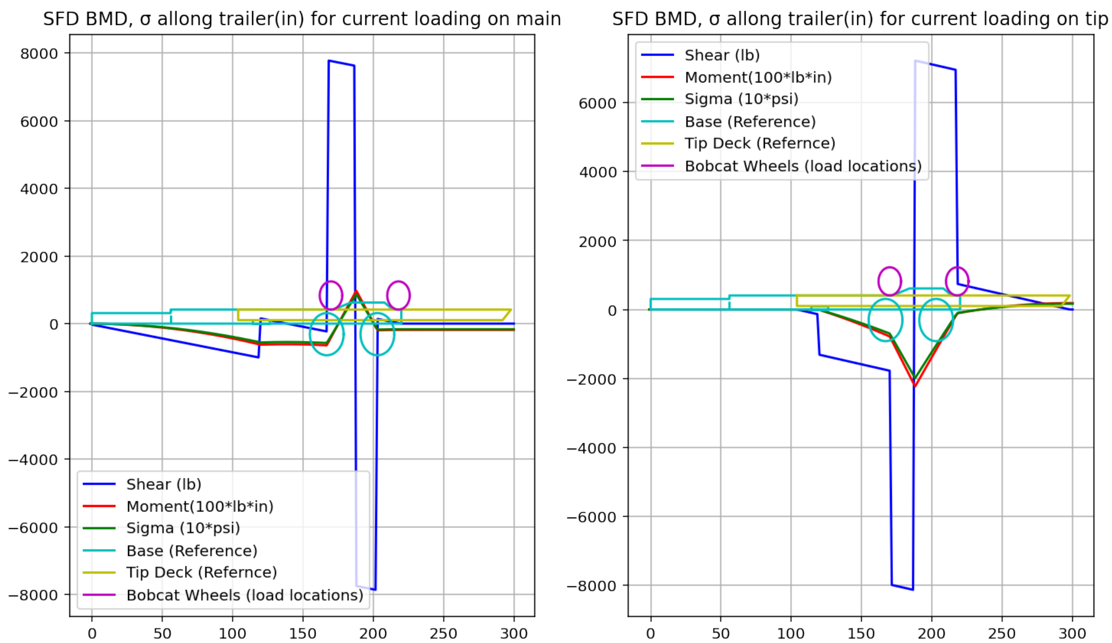
[12]:

Plots for L= 160(in) rear Load:91%

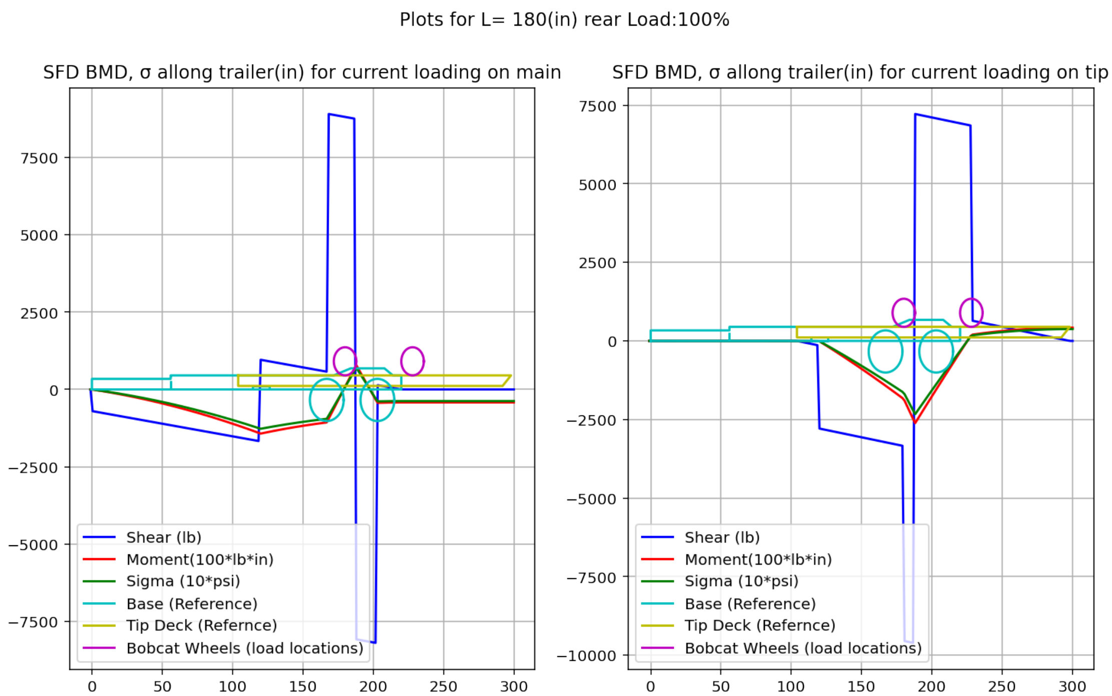


[12]:

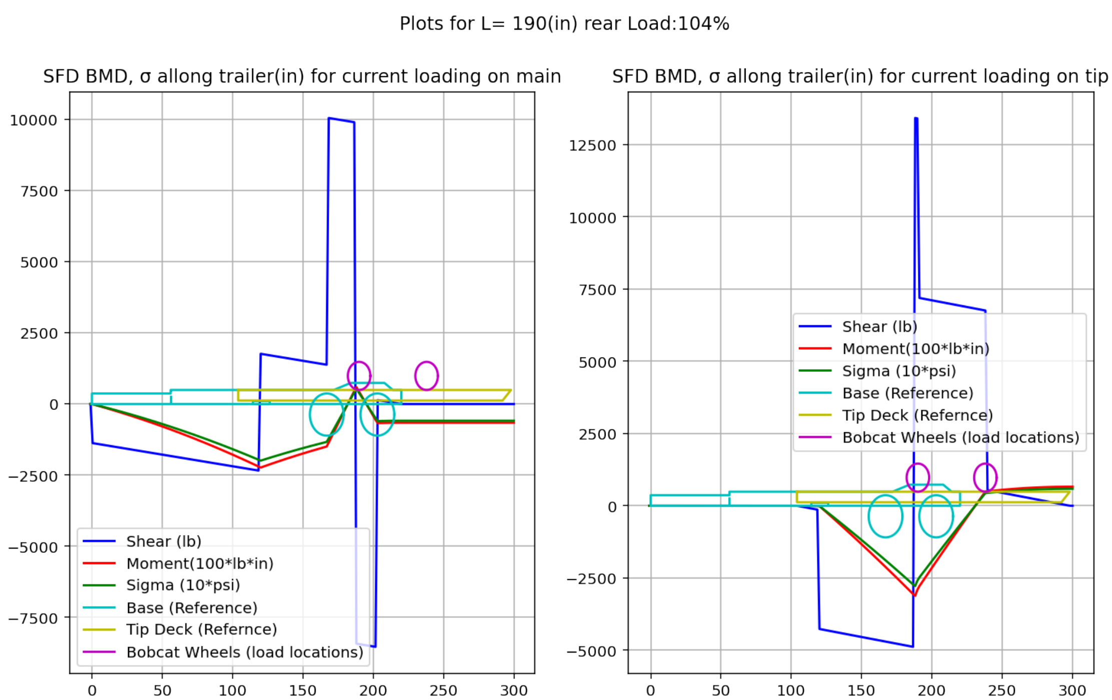
Plots for L= 170(in) rear Load:95%



[12]:

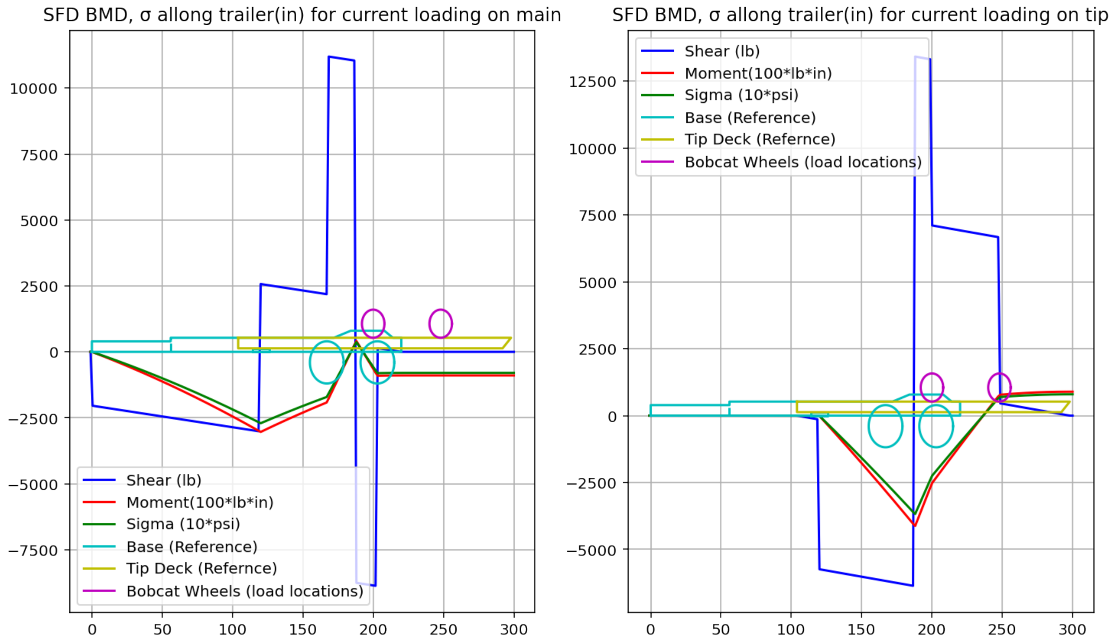

Plots for L= 180(in) rear Load:100%
SFD BMD, σ allong trailer(in) for current loading on main
SFD BMD, σ allong trailer(in) for current loading on tip
Shear (lb)
Moment(100*lb*in)
Sigma (10*psi)
Base (Reference)
Tip Deck (Refernce)
Bobcat Wheels (load locations)

[12]:


Plots for L= 190(in) rear Load:104%
SFD BMD, σ allong trailer(in) for current loading on main
SFD BMD, σ allong trailer(in) for current loading on tip
Shear (lb)
Moment(100*lb*in)
Sigma (10*psi)
Base (Reference)
Tip Deck (Refernce)
Bobcat Wheels (load locations)

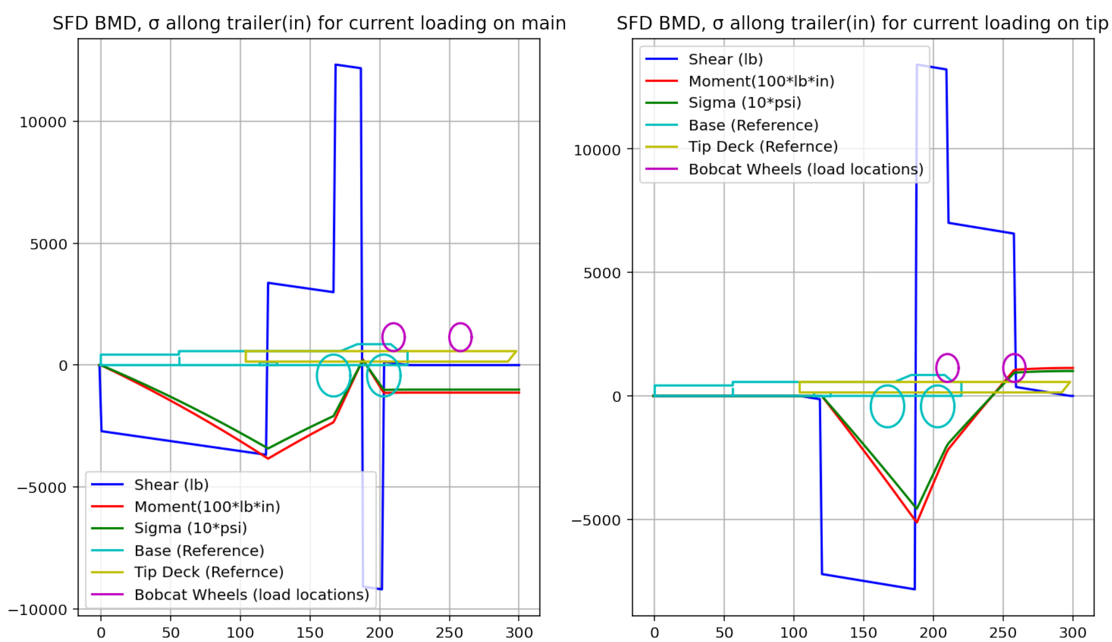
[12]:

Plots for L= 200(in) rear Load:108%



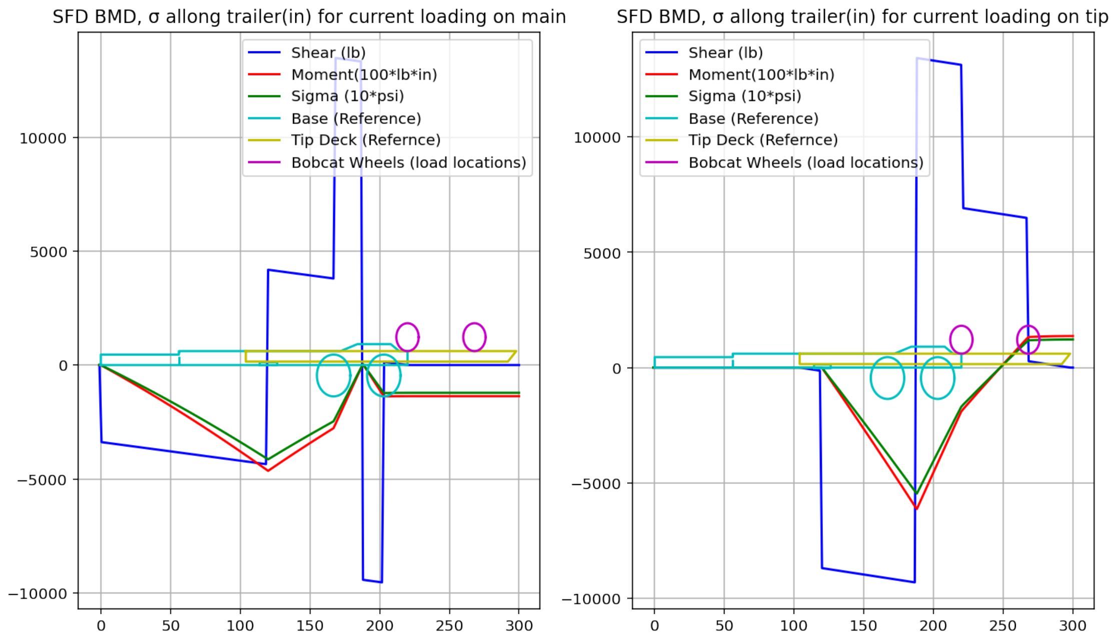
[12]:

Plots for L= 210(in) rear Load:112%



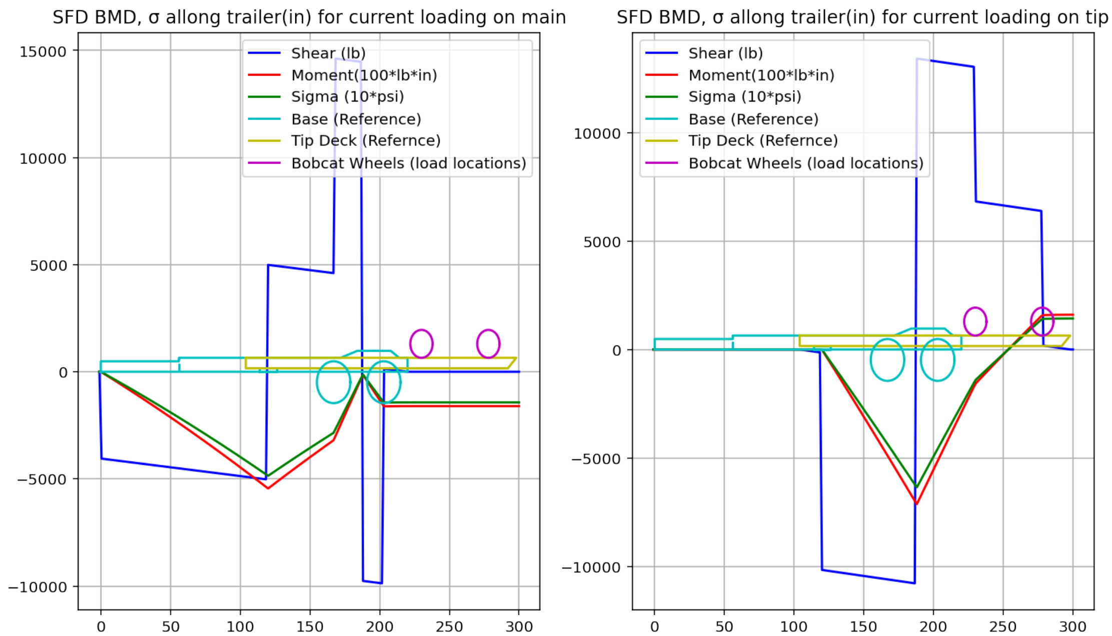
[12]:

Plots for L= 220(in) rear Load:116%

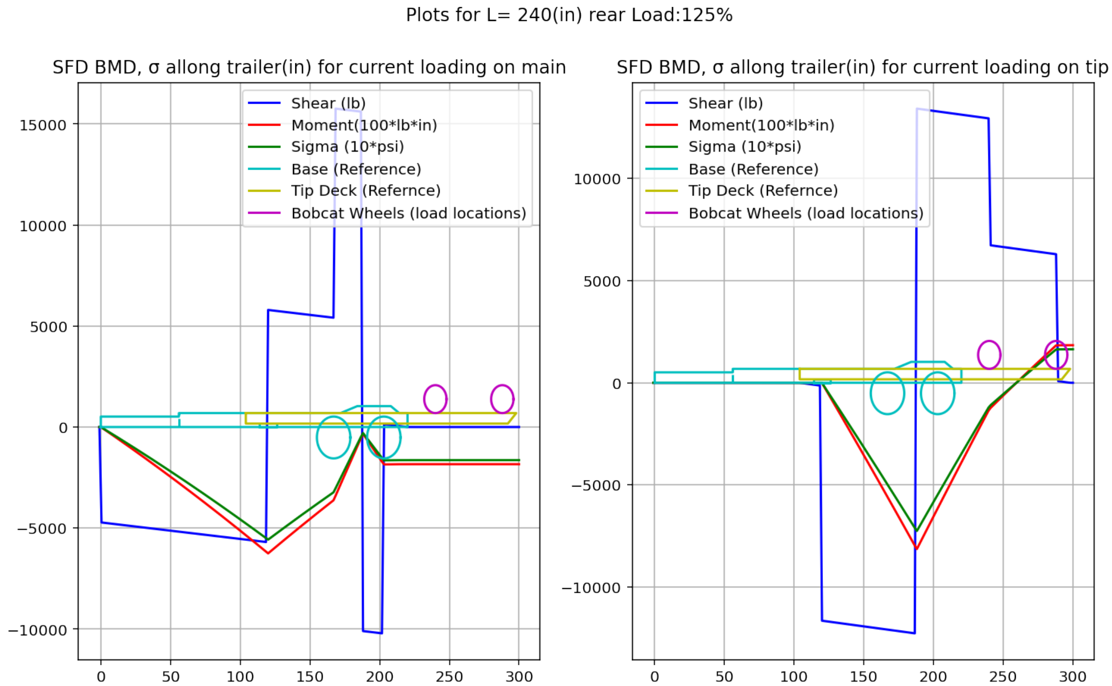


[12]:

Plots for L= 230(in) rear Load:121%



[12]:


Plots for L= 240(in) rear Load:125%
SFD BMD, σ allong trailer(in) for current loading on main
SFD BMD, σ allong trailer(in) for current loading on tip
Shear (lb)
Moment(100*lb*in)
Sigma (10*psi)
Base (Reference)
Tip Deck (Refernce)
Bobcat Wheels (load locations)

[0]:

[0]: