

LSTB November 2021 Report

November 13, 2021

1 GSF Sigma LS Geo TB Report October 2021

1.1 Load Data

```
[ ]: import pandas as pd
import math
file_paths = ['phase_1.csv', 'phase_2.csv', 'phase_3.csv', 'phase_4.csv']
data = []

for i in range(len(file_paths)):
    try:
        data.append(pd.read_csv(file_paths[i], index_col=0).
↳drop_duplicates().fillna(0))
    except:
        continue

[ ]: data[-1]['pointsPerGP'] = round(data[-1]['territoryPointsContributed'] /
                                   (data[-1]['shipGP']+data[-1]['characterGP']), 3)
avgPointsPerGP = round(data[-1]['pointsPerGP'].mean(), 3)
```

1.2 Input

1.2.1 Sandbagging

```
[ ]: tbType = 'LS'

sbag_1_top = False
sbag_1_mid = True
sbag_1_bottom = False

sbag_2_top = False
sbag_2_mid = False
sbag_2_bottom = False

sbag_3_top = True
sbag_3_mid = False
sbag_3_bottom = True
```

1.2.2 Shards and Stars

KAM shards: 32

Phase 4	Phase 3	Phase 2	Phase 1
x	3	2	2
x	1	2	3
x	3	1	3

1.3 Calculations

1.3.1 TB Points per CM

LS GEO TB

```
[ ]: if (tbType == 'LS'):  
    p1_ships_1 = [0,523900]  
    p1_ships_2 = [0,0]  
    p2_ships_2 = [0,0]  
    p3_ships_2 = [0,0]  
    p4_ships_2 = [0,0]  
  
    if(sbag_1_top):  
        p2_ships_1 = p1_ships_1  
    else:  
        p2_ships_1 = [0,900000]  
    if(sbag_2_top):  
        p3_ships_1 = p2_ships_1  
    else:  
        p3_ships_1 = [0,1800000]  
    if(sbag_3_top):  
        p4_ships_1 = p3_ships_1  
    else:  
        p4_ships_1 = [0,2750000]  
  
    p1_ground_1 = [0,403000,573500,840000,1155000]  
    p1_ground_2 = [0,403000,573500,840000,1155000]  
  
    if(sbag_1_bottom):  
        p2_ground_1 = p1_ground_1  
        p2_ground_2 = p1_ground_2  
    else:  
        p2_ground_1 = [0,434000,704000,1014750,1377000]  
        p2_ground_2 = [0,434000,704000,1014750,1377000]  
    if(sbag_2_bottom):  
        p3_ground_1 = p2_ground_1  
        p3_ground_2 = p2_ground_2  
    else:  
        p3_ground_1 = [0,464000,775500,1105000,1627500]
```

```

        p3_ground_2 = [0,464000,775500,1105000,1627500]
    if(sbag_3_bottom):
        p4_ground_1 = p3_ground_1
        p4_ground_2 = p3_ground_2
    else:
        p4_ground_1 = [0,511500,867000,1242500,1837500]
        p4_ground_2 = [0,511500,867000,1242500,1837500]

    p1_ground_3 = [0,403000,573500,840000,1155000]
    p1_ground_4 = [0,523900,745550,1092000,1501500]
    p1_ground_5 = [0,0,0,0,0]

    if(sbag_1_mid):
        p2_ground_3 = p1_ground_3
        p2_ground_4 = p1_ground_4
        p2_ground_5 = p1_ground_5
    else:
        p2_ground_3 = [0,434000,704000,1014750,1377000]
        p2_ground_4 = [0,434000,704000,1014750,1377000]
        p2_ground_5 = [0,564200,915200,1319175,1790100]
    if(sbag_2_mid):
        p3_ground_3 = p2_ground_3
        p3_ground_4 = p2_ground_4
        p3_ground_5 = p2_ground_5
    else:
        p3_ground_3 = [0,464000,775500,1105000,1627500]
        p3_ground_4 = [0,464000,775500,1105000,1627500]
        p3_ground_5 = [0,0,0,0,0]
    if(sbag_3_mid):
        p4_ground_3 = p3_ground_3
        p4_ground_4 = p3_ground_4
        p4_ground_5 = p3_ground_5
    else:
        p4_ground_3 = [0,511500,867000,1242500,1837500]
        p4_ground_4 = [0,664950,1127100,1615250,2388750]
        p4_ground_5 = [0,867000,1837500,0,0]

```

DS GEO TB

```

[ ]: if (tbType == 'DS'):

    p1_ships_1 = [0,0]
    p1_ships_2 = [0,0]
    p2_ships_1 = [0,825000]
    p2_ships_2 = [0,1072500]

    if(sbag_2_top):
        p3_ships_1 = p2_ships_1

```

```

    p3_ships_2 = p2_ships_2
else:
    p3_ships_1 = [0,1665000]
    p3_ships_2 = [0,2164500]
if(sbag_3_top):
    p4_ships_1 = p3_ships_1
    p4_ships_2 = p3_ships_2
else:
    p4_ships_1 = [0,2750000]
    p4_ships_2 = [0,0]

p1_ground_1 = [0,187500,297500,500000,792000]
p1_ground_2 = [0,187500,297500,500000,792000]

if(sbag_1_bottom):
    p2_ground_1 = p1_ground_1
    p2_ground_2 = p1_ground_2
else:
    p2_ground_1 = [0,270000,420000,708000,1080000]
    p2_ground_2 = [0,270000,420000,708000,1080000]
if(sbag_2_bottom):
    p3_ground_1 = p2_ground_1
    p3_ground_2 = p2_ground_2
else:
    p3_ground_1 = [0,336000,540000,910000,1352000]
    p3_ground_2 = [0,336000,540000,910000,1352000]
if(sbag_3_bottom):
    p4_ground_1 = p3_ground_1
    p4_ground_2 = p3_ground_2
else:
    p4_ground_1 = [0,405000,675000,1038500,1564000]
    p4_ground_2 = [0,405000,675000,1038500,1564000]

p1_ground_3 = [0,187500,297500,500000,792000]
p1_ground_4 = [0,187500,297500,500000,792000]
p1_ground_5 = [0,0,0,0,0]

if(sbag_1_mid):
    p2_ground_3 = p1_ground_3
    p2_ground_4 = p1_ground_4
    p2_ground_5 = p1_ground_5
else:
    p2_ground_3 = [0,270000,420000,708000,1080000]
    p2_ground_4 = [0,270000,420000,708000,1080000]
    p2_ground_5 = [0,351000,546000,920400,1404000]
if(sbag_2_mid):
    p3_ground_3 = p2_ground_3

```

```

        p3_ground_4 = p2_ground_4
        p3_ground_5 = p2_ground_5
    else:
        p3_ground_3 = [0,336000,540000,910000,1352000]
        p3_ground_4 = [0,336000,540000,910000,1352000]
        p3_ground_5 = [0,0,0,0,0]
    if(sbag_3_mid):
        p4_ground_3 = p3_ground_3
        p4_ground_4 = p3_ground_4
        p4_ground_5 = p3_ground_5
    else:
        p4_ground_3 = [0,405000,675000,1038500,1564000]
        p4_ground_4 = [0,405000,675000,1038500,1564000]
        p4_ground_5 = [0,1350050,2033200,0,0]

```

1.3.2 CM Points

```

[ ]: global ground_missions
ground_missions = {}
ground_missions[1] = [p1_ground_1, p1_ground_2,
                      p1_ground_3, p1_ground_4, p1_ground_5]
ground_missions[2] = [p2_ground_1, p2_ground_2,
                      p2_ground_3, p2_ground_4, p2_ground_5]
ground_missions[3] = [p3_ground_1, p3_ground_2,
                      p3_ground_3, p3_ground_4, p3_ground_5]
ground_missions[4] = [p4_ground_1, p4_ground_2,
                      p4_ground_3, p4_ground_4, p4_ground_5]

global ship_missions
ship_missions = {}
ship_missions[1] = [p1_ships_1, p1_ships_2]
ship_missions[2] = [p2_ships_1, p2_ships_2]
ship_missions[3] = [p3_ships_1, p3_ships_2]
ship_missions[4] = [p4_ships_1, p4_ships_2]

global max_ground
max_ground = {}
max_ground[1] = \
    ↪ p1_ground_1[4]+p1_ground_2[4]+p1_ground_3[4]+p1_ground_4[4]+p1_ground_5[4]
max_ground[2] = p2_ground_1[4]+p2_ground_2[4] + \
    ↪ p2_ground_3[4]+p2_ground_4[4]+p2_ground_5[4]
max_ground[3] = \
    ↪ p3_ground_1[4]+p3_ground_2[4]+p3_ground_3[4]+p3_ground_4[4]+p3_ground_5[4]
max_ground[4] = p4_ground_1[4]+p4_ground_2[4] + \
    ↪ p4_ground_3[4]+p4_ground_4[4]+p4_ground_5[2]

global max_ships

```

```

max_ships = {}
max_ships[1] = p1_ships_1[1]+p1_ships_2[1]
max_ships[2] = p2_ships_1[1]+p2_ships_2[1]
max_ships[3] = p3_ships_1[1]+p3_ships_2[1]
max_ships[4] = p4_ships_1[1]+p4_ships_2[1]

```

1.4 Low Performers

1.4.1 Lowest TB Points per GP

```

[ ]: n = 10
low_ppg = data[-1]['pointsPerGP'].sort_values().head(n)
print(low_ppg)

```

```

name
BabyYodaHitta      2.761
Zlada14            2.951
M1TTH              2.958
Maxaron Lexilon    3.467
wamakima5004       3.786
EvilCoyote2011     3.880
ONE                3.976
joker              4.336
The Wall           4.399
Theflavorgreen     4.411
Name: pointsPerGP, dtype: float64

```

1.4.2 Lowest CM Waves Completed

```

[ ]: low_cm = data[-1]['combatMissionWavesCompleted'].sort_values().head(n)
print(low_cm)

```

```

name
M1TTH      6
Theflavorgreen  6
BabyYodaHitta  7
joker      12
Higgs      12
EvilCoyote2011  13
Zhil Axflow  14
Maxaron Lexilon  14
ONE           16
Zlada14       17
Name: combatMissionWavesCompleted, dtype: int64

```

1.4.3 Lowest TB Points

```
[ ]: low_tb = data[-1]['territoryPointsContributed'].sort_values().head(n)
      print(low_tb)
```

```
name
BabyYodaHitta      8241802
M1TTH              13311680
Zlada14            17943482
Dark Penguin      20763171
Larping Soccer Moms 21153200
MINI xipokemastrix 22231858
joker              22571667
Obi Won Sebroni    22744529
The Wall           22872153
Maxaron Lexilon    22949620
Name: territoryPointsContributed, dtype: int64
```

1.5 Top Performers

1.5.1 Highest TB Points per GP

```
[ ]: high_ppg = data[-1]['pointsPerGP'].sort_values(ascending = False).head(n)
      print(high_ppg)
```

```
name
PadawanTano      8.667
Loadage          7.639
MINI Stewabob    7.636
Revanche Gilder  7.490
Tommyboy85       7.375
Exeel            7.243
s o l o          7.018
Agave            6.915
ilekkund2        6.803
OttoVonGens      6.740
Name: pointsPerGP, dtype: float64
```

1.5.2 Highest Combat Waves Completed

```
[ ]: high_cm = data[-1]['combatMissionWavesCompleted'].sort_values(
      ascending=False).head(n)
      print(high_cm)
```

```
name
s o l o          61
Tommyboy85       56
Loadage          55
Elros Halfelven  52
Exeel            52
```

```

OttoVonGens      51
Revanche Gilder  51
Flywire          49
Chaunce          48
Calens           47
Name: combatMissionWavesCompleted, dtype: int64

```

1.5.3 Highest TB Points

```

[ ]: high_tb = data[-1]['territoryPointsContributed'].sort_values(
      ascending=False).head(n)
      print(high_tb)

```

```

name
s o l o      55775719
Tommyboy85    48230989
OttoVonGens   47940257
ilekkund      47214174
Exeel         46811390
Loadage       45410212
Heywood Jablowme 44793000
Revanche Gilder 43708128
Komitadji Halfelven 42774984
Chaunce       41786505
Name: territoryPointsContributed, dtype: int64

```

1.6 Guild Performance

```

[ ]: def toPoints(points,waves):
      i = 0
      point_value = pd.Series([0]*len(waves),index = waves.index)
      while(i<len(waves)):
          point_value[i] = points[waves.iloc[i].astype('int64')]
          i+=1
      return point_value

```

```

[ ]: def percents(df,phase):
      if(not ("Ch 5") in df.columns):
          df['Ch 5'] = [0]*len(df)

      if(not ("Fl 1") in df.columns):
          df['Fl 1'] = [0]*len(df)

      if(not ("Fl 2") in df.columns):
          df['Fl 2'] = [0]*len(df)

```



```

df['ground'] = toPoints(ground_missions[phase][0], df['Ch_
→1'])+toPoints(ground_missions[phase][1], df['Ch_
→2'])+toPoints(ground_missions[phase][2], df['Ch_
→3'])+toPoints(ground_missions[phase][3], df['Ch_
→4'])+toPoints(ground_missions[phase][4], df['Ch 5'])

df['ship'] = toPoints(ship_missions[phase][0], df['Fl 1'])+ \
toPoints(ship_missions[phase][1], df['Fl 2'])

df['ground_perc'] = round((df['ground']/max_ground[phase])*100, 0)

df['ship_perc'] = round((df['ship']/max_ships[phase])*100, 0)

ground_perc = df['ground_perc'].mean()
ship_perc = df['ship_perc'].mean()

# ground_perc = round(
#     pd.Series(df['ground']/max_ground[phase]).mean()*100, 0)

# ship_perc = round(
#     pd.Series(df['ship']/max_ships[phase]).mean()*100, 0)

return (ground_perc, ship_perc)

```

1.6.1 Percent of Combat Mission Points per Phase

```

[ ]: perc = list()
for i in range(len(data)):
    (ground_perc, ship_perc) = percents(data[i], i + 1)
    if math.isnan(ship_perc):
        ship_perc = 0
    if math.isnan(ground_perc):
        ground_perc = 0

    perc.append([str(ground_perc) + '%', str(ship_perc) + '%'])

for i in range(len(data), 4):
    perc.append(['NA', 'NA'])

perc_points = pd.DataFrame(perc, index=[
    'Phase 1', 'Phase 2', 'Phase 3', 'Phase 4'],
→columns=['Ground', 'Ships'])
print(perc_points)

```

	Ground	Ships
Phase 1	45.78%	38.0%
Phase 2	46.72%	50.0%

Phase 3 45.12% 0.0%
Phase 4 27.18% 48.0%

1.6.2 Percent of CM Completed by Player

```
[ ]: player_perc = pd.DataFrame()
for i in range(len(data)):
    ground_name = 'P' + str(i+1) + 'G'
    ship_name = 'P' + str(i+1) + 'S'
    player_perc[ground_name] = data[i]['ground_perc']
    player_perc[ship_name] = data[i]['ship_perc']
print(player_perc)
```

	P1G	P1S	P2G	P2S	P3G	P3S	P4G	P4S
name								
Obi Won Sebroni	0.0	0.0	33.0	0.0	24.0	0.0	17.0	0.0
Gryphix	68.0	0.0	46.0	100.0	31.0	0.0	20.0	100.0
Wolfman314	0.0	0.0	75.0	100.0	37.0	0.0	22.0	100.0
Zhil Axfow	34.0	100.0	15.0	0.0	19.0	0.0	13.0	0.0
Revanche Gilder	81.0	100.0	62.0	100.0	59.0	0.0	75.0	100.0
Exeel	77.0	0.0	87.0	100.0	100.0	0.0	43.0	100.0
Agave	59.0	100.0	60.0	100.0	74.0	0.0	40.0	100.0
Loadage	85.0	100.0	94.0	100.0	69.0	0.0	59.0	100.0
ilekkund	92.0	100.0	38.0	100.0	50.0	0.0	55.0	100.0
Theflavorgreen	0.0	0.0	13.0	0.0	7.0	0.0	0.0	0.0
ilekkund2	52.0	100.0	53.0	100.0	50.0	0.0	47.0	0.0
MINI xipokemastrix	0.0	0.0	33.0	100.0	25.0	0.0	22.0	0.0
Flywire	94.0	100.0	49.0	0.0	87.0	0.0	49.0	0.0
TacoPizza	34.0	100.0	51.0	100.0	25.0	0.0	8.0	0.0
Maxaron Lexilon	0.0	0.0	37.0	100.0	0.0	0.0	12.0	0.0
Neeb	34.0	0.0	41.0	0.0	19.0	0.0	12.0	0.0
Sultan2309	40.0	0.0	41.0	0.0	50.0	0.0	26.0	0.0
PadawanTano	57.0	100.0	47.0	0.0	49.0	0.0	48.0	100.0
Doomslug the Destroyer	35.0	0.0	41.0	0.0	25.0	0.0	20.0	0.0
Larping Soccer Moms	27.0	100.0	36.0	0.0	25.0	0.0	31.0	100.0
JustinAlexander11	67.0	0.0	54.0	0.0	44.0	0.0	17.0	100.0
Elros Halfelven	85.0	100.0	76.0	100.0	79.0	0.0	59.0	0.0
Philo Beddoe	47.0	0.0	0.0	0.0	59.0	0.0	17.0	100.0
BabyYodaHitta	17.0	0.0	25.0	0.0	0.0	0.0	0.0	0.0
s o l o	100.0	100.0	94.0	100.0	92.0	0.0	61.0	100.0
Higgs	0.0	0.0	20.0	0.0	17.0	0.0	0.0	0.0
ShootMeow	50.0	100.0	0.0	100.0	50.0	0.0	49.0	100.0
Calens	77.0	0.0	62.0	0.0	100.0	0.0	43.0	0.0
ONE	34.0	0.0	36.0	0.0	24.0	0.0	5.0	0.0
Promethean	92.0	100.0	61.0	100.0	54.0	0.0	43.0	0.0
Guntha Arbos	0.0	0.0	42.0	0.0	12.0	0.0	23.0	0.0
The Wall	30.0	0.0	43.0	100.0	37.0	0.0	17.0	0.0
AKB	87.0	0.0	0.0	0.0	67.0	0.0	26.0	100.0

Baxston Kane	0.0	0.0	66.0	100.0	54.0	0.0	17.0	100.0
Komitadji Halfelven	0.0	0.0	86.0	100.0	54.0	0.0	35.0	100.0
Argarax	70.0	0.0	49.0	100.0	42.0	0.0	17.0	100.0
Dark Penguin	50.0	0.0	24.0	0.0	12.0	0.0	8.0	100.0
SloppySaberFlavor	73.0	0.0	87.0	0.0	50.0	0.0	29.0	0.0
MINI Stewabob	39.0	100.0	54.0	100.0	42.0	0.0	26.0	100.0
Tommyboy85	77.0	100.0	87.0	100.0	84.0	0.0	63.0	100.0
Heywood Jablowme	55.0	0.0	54.0	100.0	75.0	0.0	43.0	100.0
Zlada14	68.0	100.0	36.0	0.0	0.0	0.0	0.0	0.0
Chaunce	87.0	100.0	79.0	100.0	92.0	0.0	17.0	100.0
Plucky Haydon	0.0	0.0	69.0	100.0	59.0	0.0	55.0	0.0
joker	0.0	0.0	0.0	0.0	36.0	0.0	8.0	0.0
OttoVonGens	85.0	0.0	94.0	100.0	100.0	0.0	17.0	100.0
wamakima5004	62.0	100.0	21.0	0.0	42.0	0.0	17.0	0.0
EvilCoyote2011	0.0	0.0	29.0	0.0	12.0	0.0	8.0	0.0
Masajj Vemtits	68.0	0.0	36.0	0.0	42.0	0.0	20.0	100.0
M1TTH	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

1.6.3 Average TB Points per GP

```
[ ]: print(avgPointsPerGP)
```

5.606

1.6.4 Guild TB Points and TB Points per GP

```
[ ]: data[-1].loc[:,['territoryPointsContributed','pointsPerGP']].
      ↪sort_values(by=['territoryPointsContributed'], ascending=False)
```

	territoryPointsContributed	pointsPerGP
name		
s o l o	55775719	7.018
Tommyboy85	48230989	7.375
OttoVonGens	47940257	6.740
ilekkund	47214174	6.391
Exeel	46811390	7.243
Loadage	45410212	7.639
Heywood Jablowme	44793000	6.343
Revanche Gilder	43708128	7.490
Komitadji Halfelven	42774984	6.476
Chaunce	41786505	6.355
Agave	41551308	6.915
Baxston Kane	40605989	6.097
Elros Halfelven	40028198	6.372
Calens	39515485	6.334
ShootMeow	39380240	6.027
Gryphix	36288665	5.346
Promethean	36061371	6.418

Masajj Vemtits	35687274	5.903
JustinAlexander11	35634369	6.134
AKB	35535518	5.936
ilekkund2	34378928	6.803
Philo Beddoe	33596041	5.673
Flywire	33396803	6.013
SloppySaberFlavor	32878511	5.516
Argarax	32495412	5.858
Plucky Haydon	31752563	5.238
Wolfman314	30703220	4.856
Guntha Arbos	29882470	4.725
Sultan2309	28260040	6.108
Zhil Axflow	28028017	4.423
Neeb	27616165	4.725
MINI Stewabob	27510012	7.636
PadawanTano	26487625	8.667
Higgs	26328298	4.712
ONE	25363779	3.976
Doomslug the Destroyer	24692479	5.648
TacoPizza	23617737	4.456
EvilCoyote2011	23356358	3.880
Theflavorgreen	23304075	4.411
wamakima5004	23287303	3.786
Maxaron Lexilon	22949620	3.467
The Wall	22872153	4.399
Obi Won Sebroni	22744529	5.764
joker	22571667	4.336
MINI xipokemastrix	22231858	5.144
Larping Soccer Moms	21153200	6.075
Dark Penguin	20763171	4.800
Zlada14	17943482	2.951
M1TTH	13311680	2.958
BabyYodaHitta	8241802	2.761