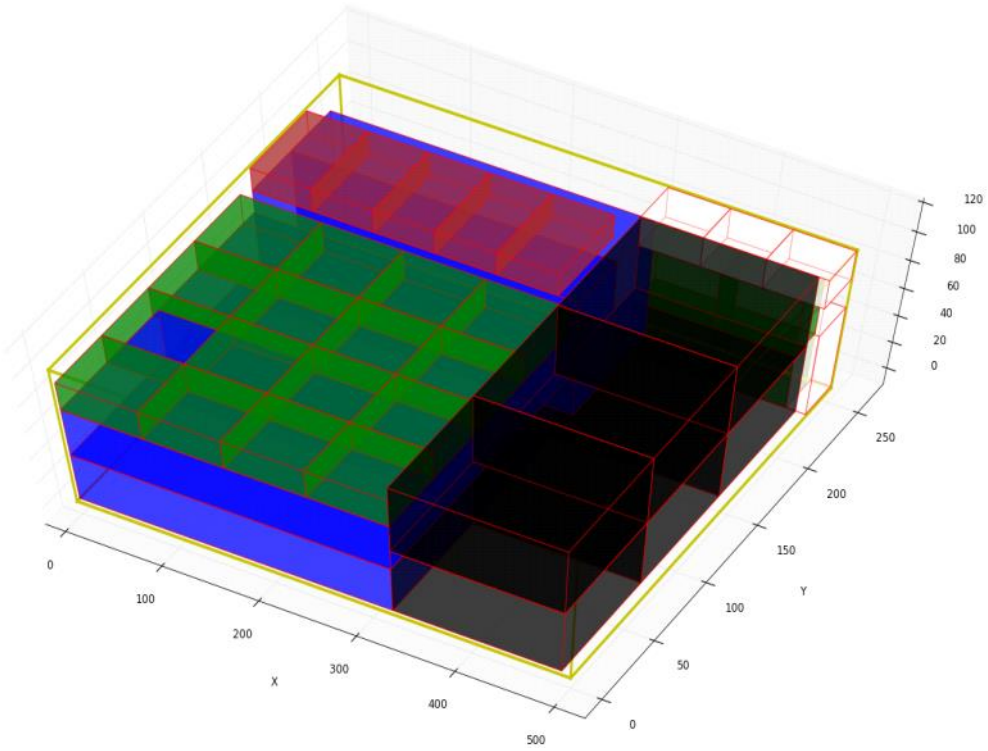
後車箱擺放計算

Friday, January 18, 2019 11:56 AM

問題:

車箱擺放大小不同行李箱的狀況(例:XXL, XL, L, M, S...)



思考:

1. 擬人法 : 模擬人的擺法,長邊先擺,靠邊擺,大的箱子先擺...
2. 貪心法: 優先擺大箱子(事實上擺小箱子可得最大填充率),箱子最長設為X方向,其次Y,其次Z
3. 可放置點: 每個放置點上測試箱子的每個方向是否可放入(考慮放入最省空間位置由基線控制?)
4. 靠邊置放
5. 砌磚法: 强制X方向擺完才能擺Y方向,整個'平面'擺完才能昇高Z擺放
6. 模擬退火法: 最佳化(目前未考慮)
7. 改變順序: XXL->XL->L... => L->XXL->S...
8. 改變方向: L.W.H -> W.H.L...

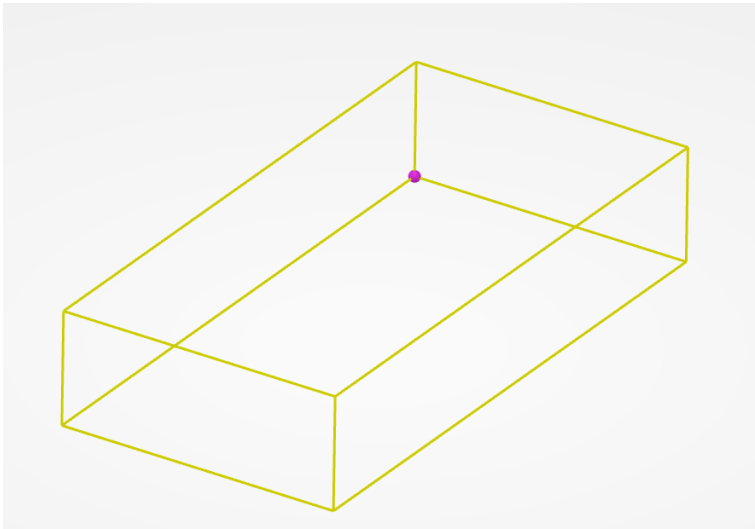
程式:

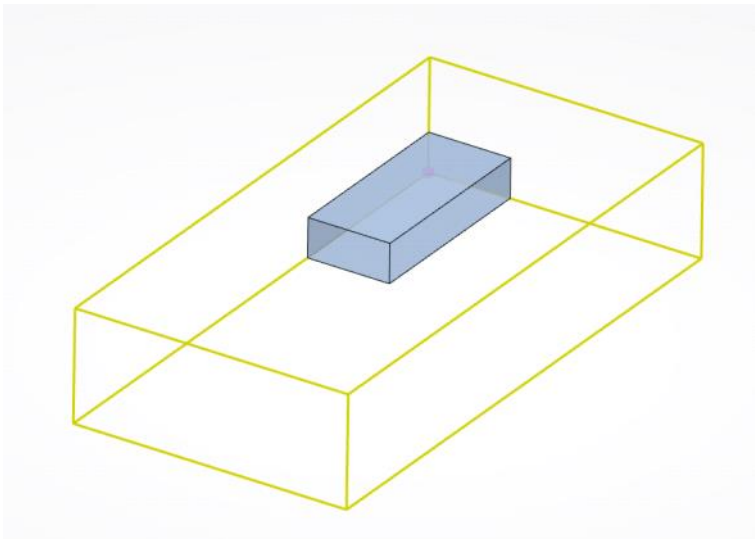
Gui_BP.py
DrawBox.py
Packing3D.py
MyMath.py

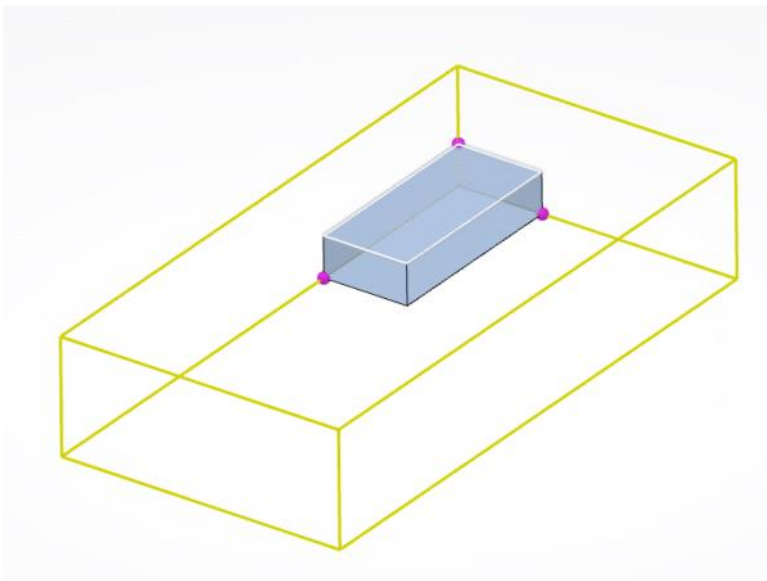
說明:

1. 可放置點:

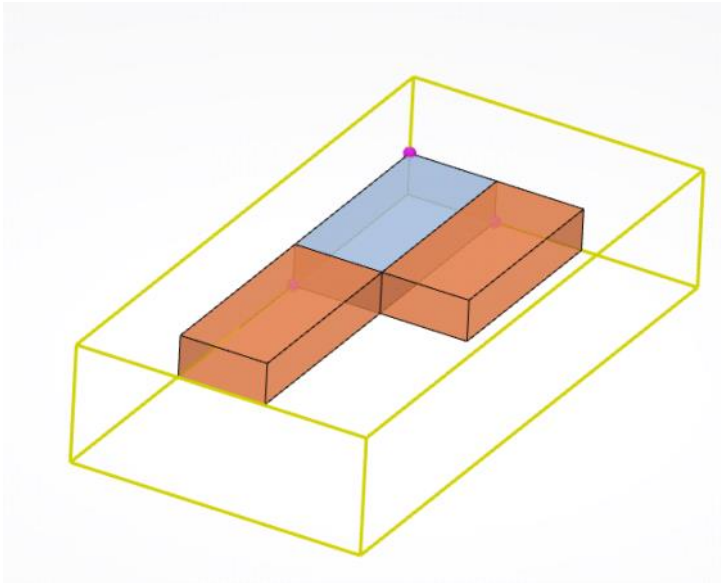
   如何增加/減少點



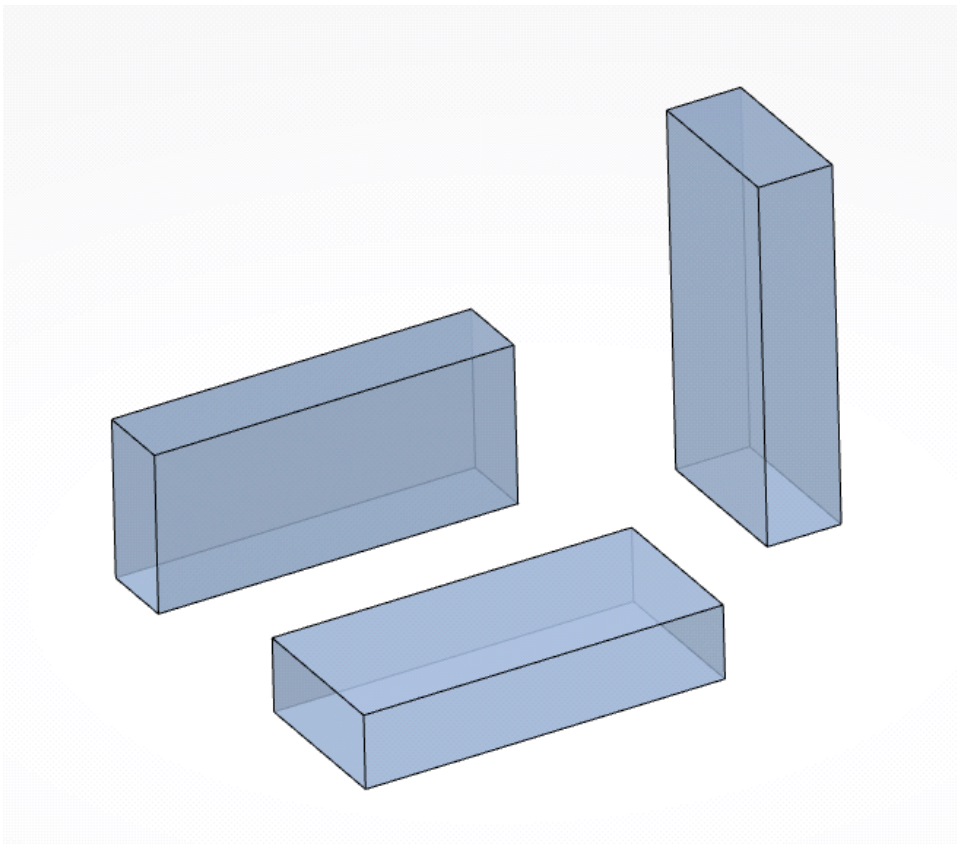




Z->X->Y 優先



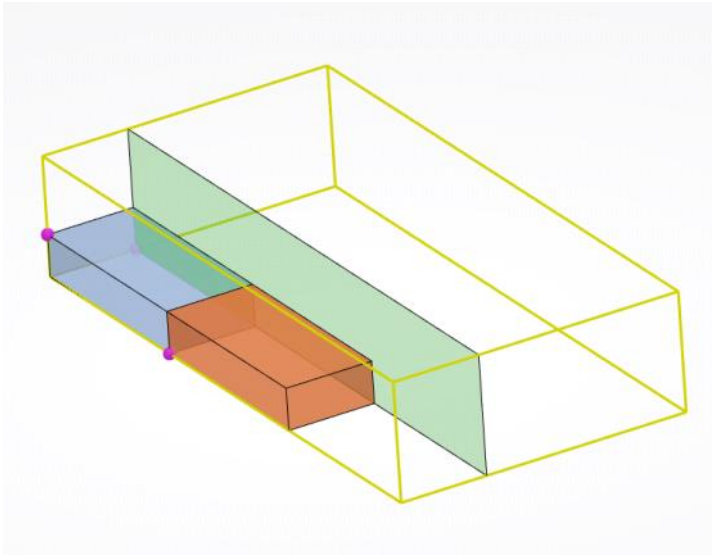
2. 箱子所有可能擺放方向(考濾24種)

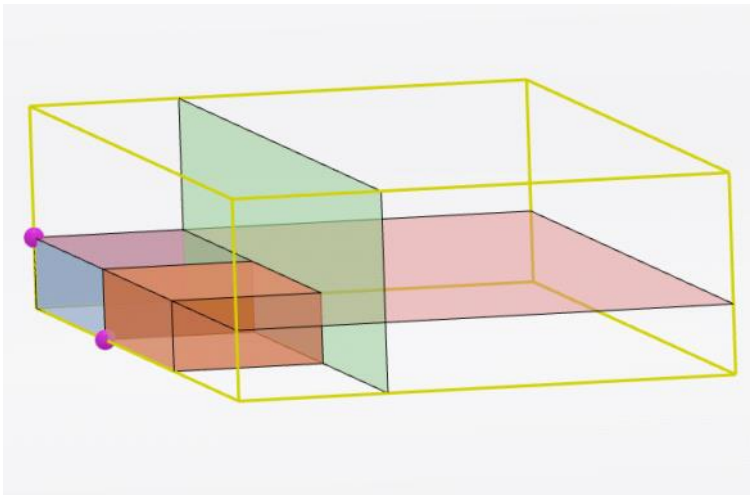


```
def set_box_dir(self,i): # 箱子不同擺放方向
if i==0:
self.r=[self.l,self.w,self.h]
if i==1:
self.r=[self.l,self.h,self.w]
if i==2:
self.r=[self.w,self.l,self.h]
if i==3:
self.r=[self.w,self.h,self.l]
if i==4:
```

```
self.r=[self.h,self.l,self.w]
if i==5:
self.r=[self.h,self.w,self.l]
if i==6:
self.r=[self.l,-self.w,self.h]
if i==7:
self.r=[self.l,-self.h,self.w]
if i==8:
self.r=[self.w,-self.l,self.h]
if i==9:
self.r=[self.w,-self.h,self.l]
if i==10:
self.r=[self.h,-self.l,self.w]
if i==11:
self.r=[self.h,-self.w,self.l]
if i==12:
self.r=[-self.l,self.w,self.h]
if i==13:
self.r=[-self.l,self.h,self.w]
if i==14:
self.r=[-self.w,self.l,self.h]
if i==15:
self.r=[-self.w,self.h,self.l]
if i==16:
self.r=[-self.h,self.l,self.w]
if i==17:
self.r=[-self.h,self.w,self.l]
if i==18:
self.r=[-self.l,-self.w,self.h]
if i==19:
self.r=[-self.l,-self.h,self.w]
if i==20:
self.r=[-self.w,-self.l,self.h]
if i==21:
self.r=[-self.w,-self.h,self.l]
if i==22:
self.r=[-self.h,-self.l,self.w]
if i==23:
self.r=[-self.h,-self.w,self.l]
```

3. 基線: 確保箱子->X->Y->Z放置





4. 干涉檢查:

   排除長方體所有不相交的狀況後,其他即為干涉