

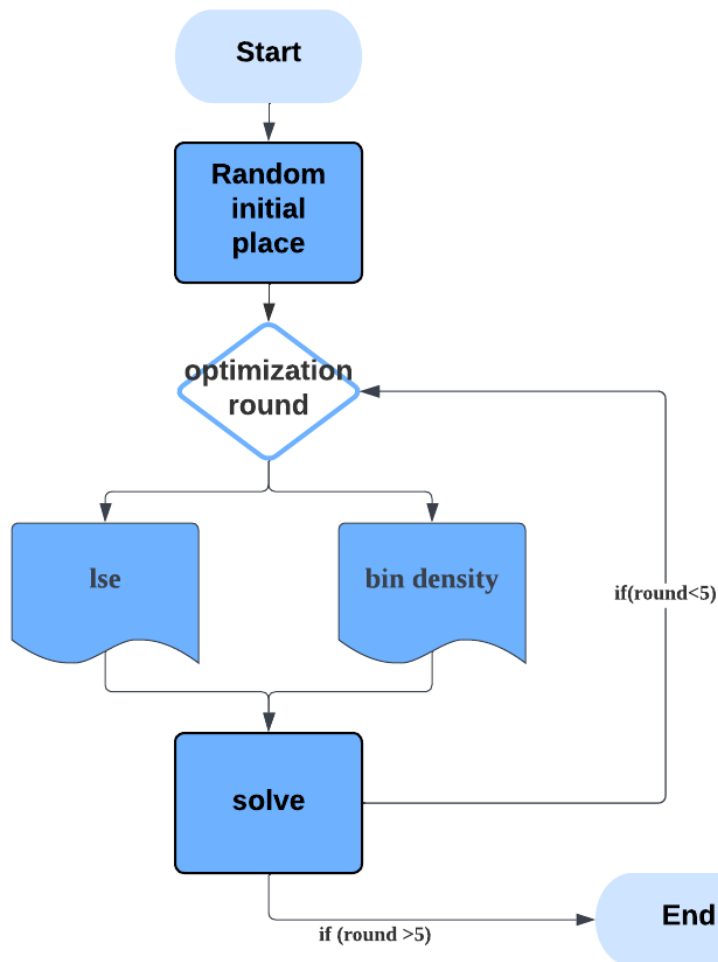
HW4 Report

111065519 林詒珊

1. The wirelength and the runtime of each testcase

	Public1	Public2	Public3
wirelength	69214968	11980715	458542175
runtime (s)	39.89	90.19	317.27

2. The details of your algorithm



我使用lse + bin density來做為優化global placement的方式，執行步驟如下：

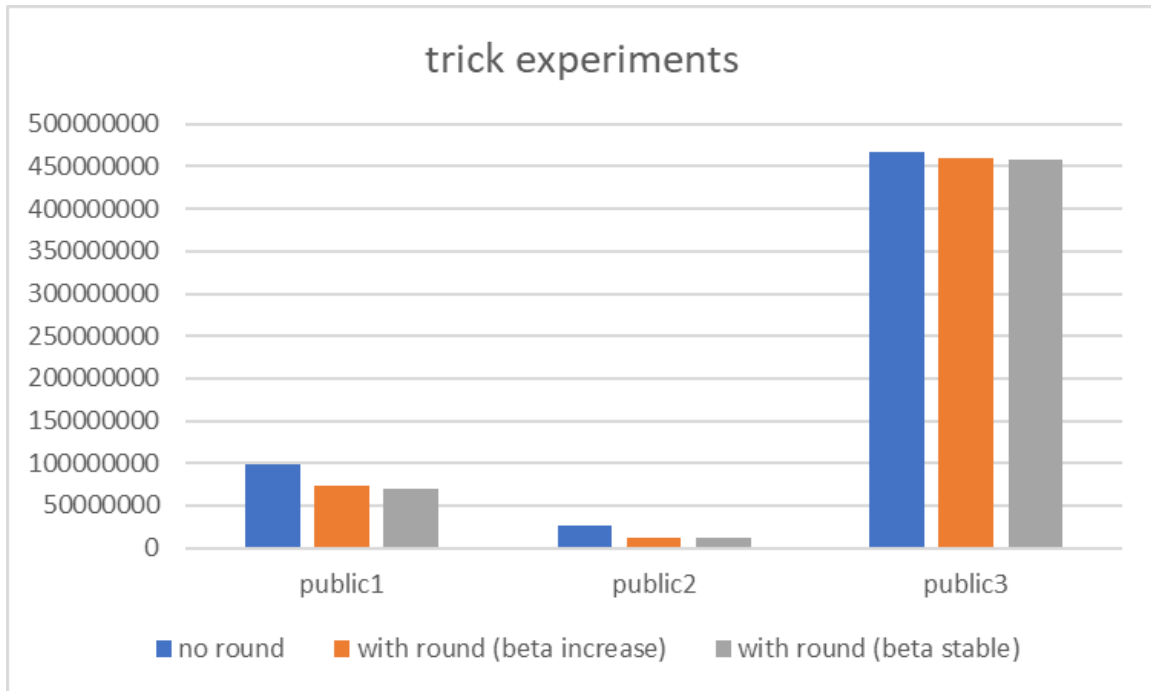
1. random出一個初始placement
2. 由初始placement的結果來計算lse 和bin density的objective function，如下

$$: \sum_{e \in E} c_e \times WL_e(\mathbf{x}, \mathbf{y}) + \beta \times \sum_b (D_b(\mathbf{x}, \mathbf{y}) - T_b)^2$$

3. 計算出objective function的微分
4. 根據gradient更新placement的位置
5. 若round還沒結束會對beta做增加，然後拿上一輪的結果繼續optimize，來去加強module的分布均勻度

3. Solution quality enhance

1. 我一開始是只會跑完一次optimization就結束global placement，但後來想到老師上課有說到可以遞增beta，所以我就設計成好幾個round，一開始beta會是0，然後逐round遞增，我發現這樣的方法在有些case 可以進步很多，但因為到後面beta太大可能讓bin density的限制蓋過wl的限制所以WL並沒有如預期般下降，因此我將round改成一開始beta為0，後面的beta就維持定直重複做iteration，實驗結果如下圖所示：



2. tune 參數: 這次作業的case有很大的差異，他們都各自可以找到一組不錯的參數，比如說bin number或是round number和beta，我認為這些跟core本身的大小有一些關聯，因此我將程式設定成如果core大到一定程度的話我會切多一些bin或者是core比較小我就會將beta設定的小一點，加強lse這個部分的loss 大小，只不過我只是根據三種不同的testcase來做猜測，因此我也不能保證我的推論是否正確，只能等之後拿到private看看結果來驗證了

4. Compare my results with the previous top 5 students' results

public1: 我在public1的結果有接近rank 1的同學

public2: publi2我有接近到wl最高的同學

public3: public3我可以接近到第二低的結果

我認為我的特點在於我的參數會根據一些猜測到的特徵做調整，所以最後的wirelength可以達到還可以的結果，但因為我的結果在public2這個case與前五名還有不小的差距，所以我在想為不會是因為我的算法只包含lse和bin density constrain，或許未來可以試試加入其他的限制進入objective function中，看看能不能達到更優的結果