TimberWolf Hierarchical Placement Algorithm

ECE 6133, Spring 2019

Bishoy Metry, Jaison George

Introduction

- Placement is the process of finding optimal physical locations to place partitioned blocks or cells on a bounded surface
- TimberWolfSC v 7.0 uses a probabilistic, iterative optimization technique, based on the simulated annealing algorithm
- It uses hierarchical clustering and placement of standard cells to achieve further minimization in wire length of standard cell placement, using simulated annealing

Problem Formulation

- Our objective is to minimize overall wirelength after placement.

- The main constraint is avoiding cells’ overlap

Algorithm Discussion

- First and second level clustering
  - Standard cells are clustered into first level clusters based on the distribution pins of the set of nets connected to randomly selected cell
  - First level clusters are randomly selected and grouped into second level clusters in the same manner that cells were added to first level clusters
  - The probability of moving a cell or first level cluster to a higher level cluster is based on the simulated annealing algorithm
  - The cost function used for clustering is depended on the fanout of the nets across target level clusters.

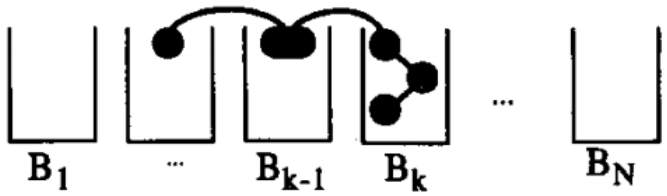


Figure 2. First level clustering [1]

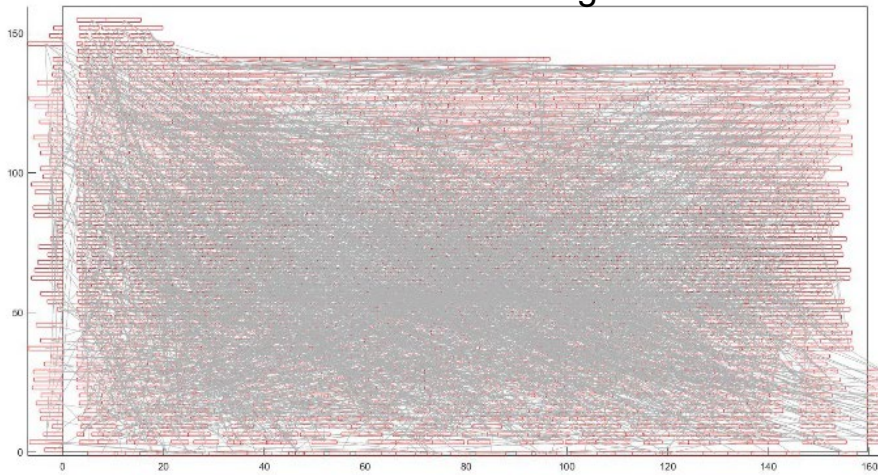
Algorithm Discussion (continued)

- Placement
  - A random cell a,a random row r, and a position x in r where cell a is to be placed are generated. Row capacity constraints are checked.
  - If capacity is not violated, cost function is calculated for this move.
  - If capacity would be violated, propose a swap with cell b present at location x in r. Capacity constraints for both rows are checked, and the cost function is calculated if there are no violations
  - Annealing is used to check if move will be accepted or not: random < exp(-Delta-Cost/T))
  - Exit conditions: Freezing point of temperature, or, Min. Acceptance Rate reached, or Time limit reached.
  - Cost function determined by change in wire lengths of cell being moved, and cells shifted
- Hierarchical Placement
  - Same logic of placement, but with level 2 clusters, level 1 clusters, then cells.

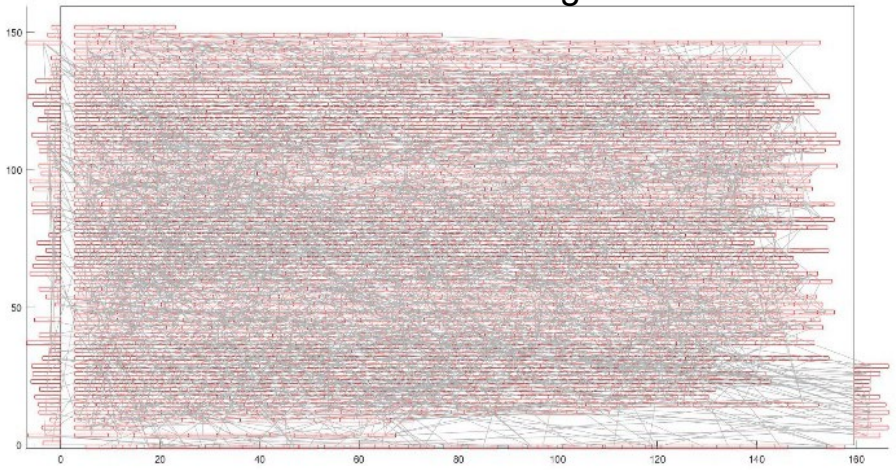
Experimental Results

StructP

Before Annealing



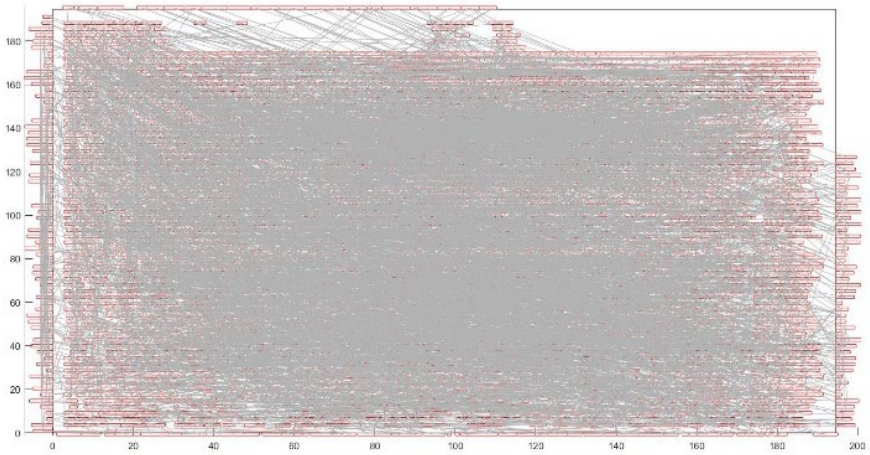
After Annealing



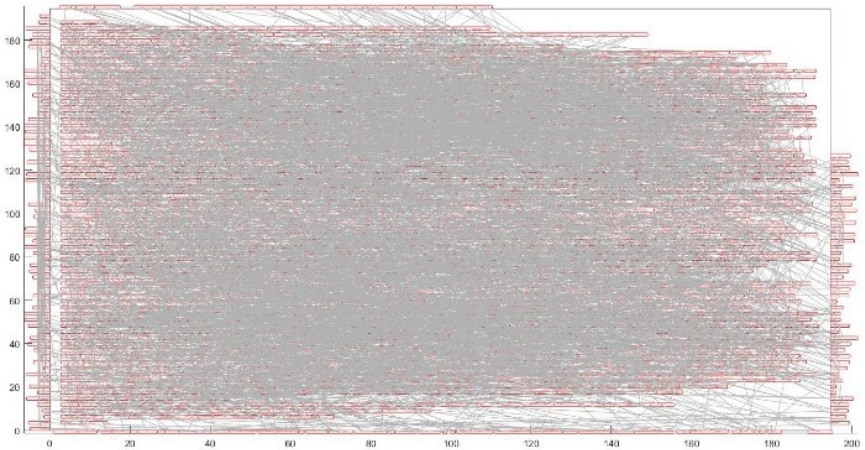
The total wire length went down from 137,881.00 um to 53,367.20 um. Final rows' widths are balanced on average. There is no overlap anywhere and the IO cells are placed perfectly around the chip border.

P2

Before Annealing



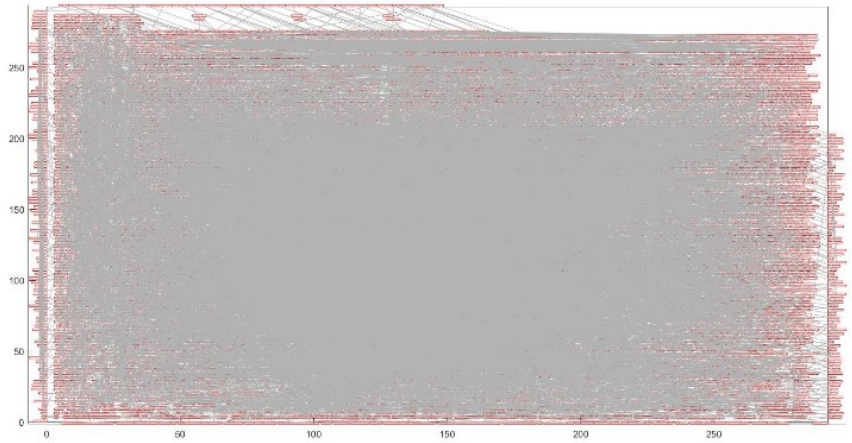
After Annealing



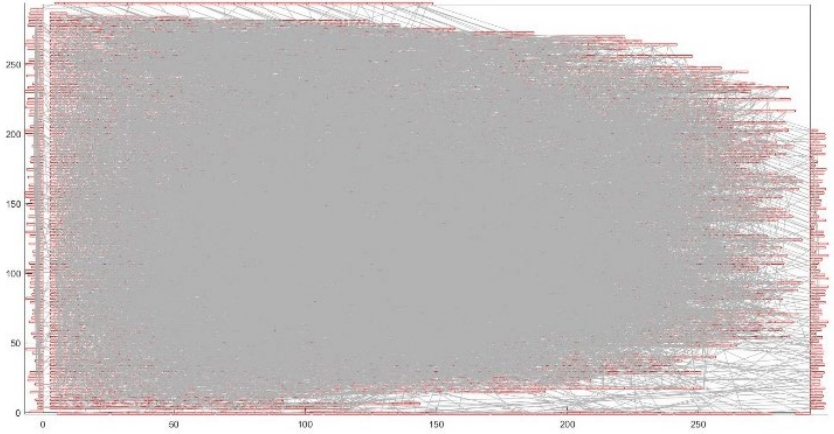
Wire length went down from 326,839.00 um to 175,578.00 um. Final rows' widths are balanced on average. There is no overlap anywhere and the IO cells are placed perfectly around the chip border.

biomedP

Before Annealing



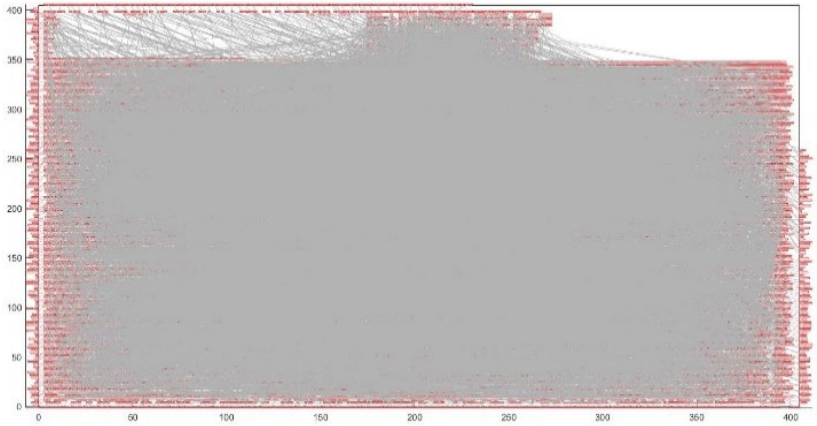
After Annealing



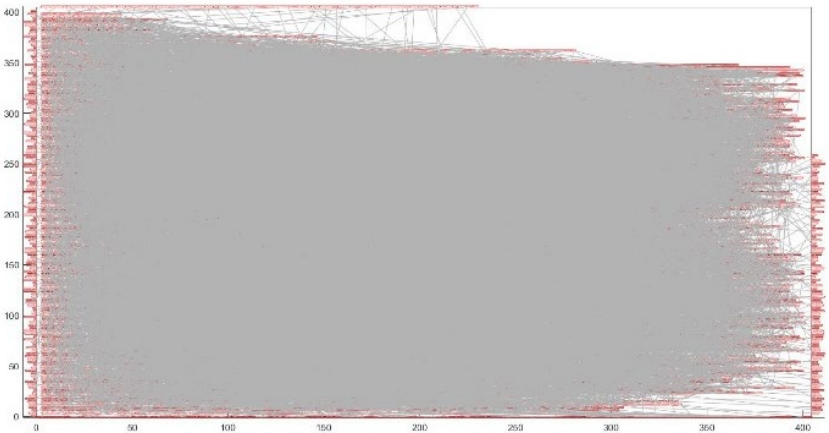
Wire length went down from 974,706.00 um to 476,999.00 um. Final rows' widths are balanced on average. There is no overlap anywhere and the IO cells are placed perfectly around the chip border.

industry2

Before Annealing



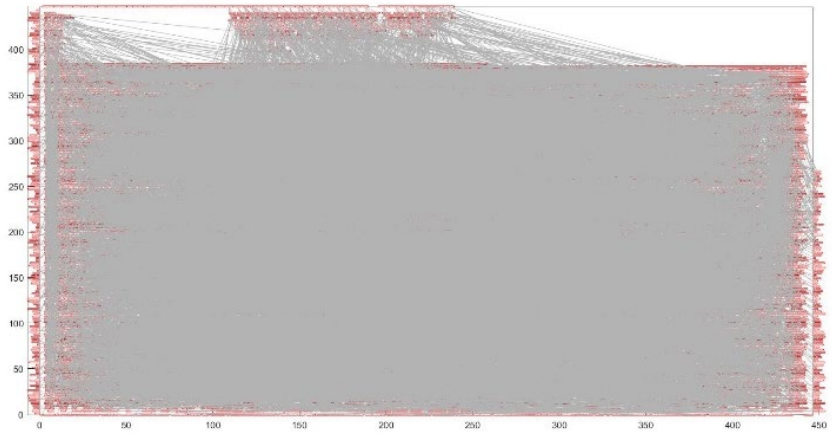
After Annealing



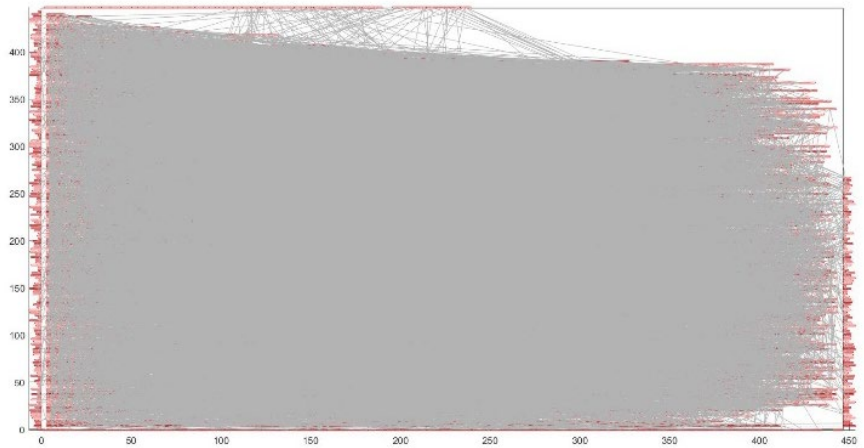
Wire length went down from 2,646,960.00 um to 1,538,010.00 um. Initially there was a number of cells on their own at the top that were placed in the initial placement close to an IO there. After annealing, we see that we have a better row-width balance. There is no overlap anywhere and the IO cells are placed perfectly around the chip border.

industry3

Before Annealing



After Annealing

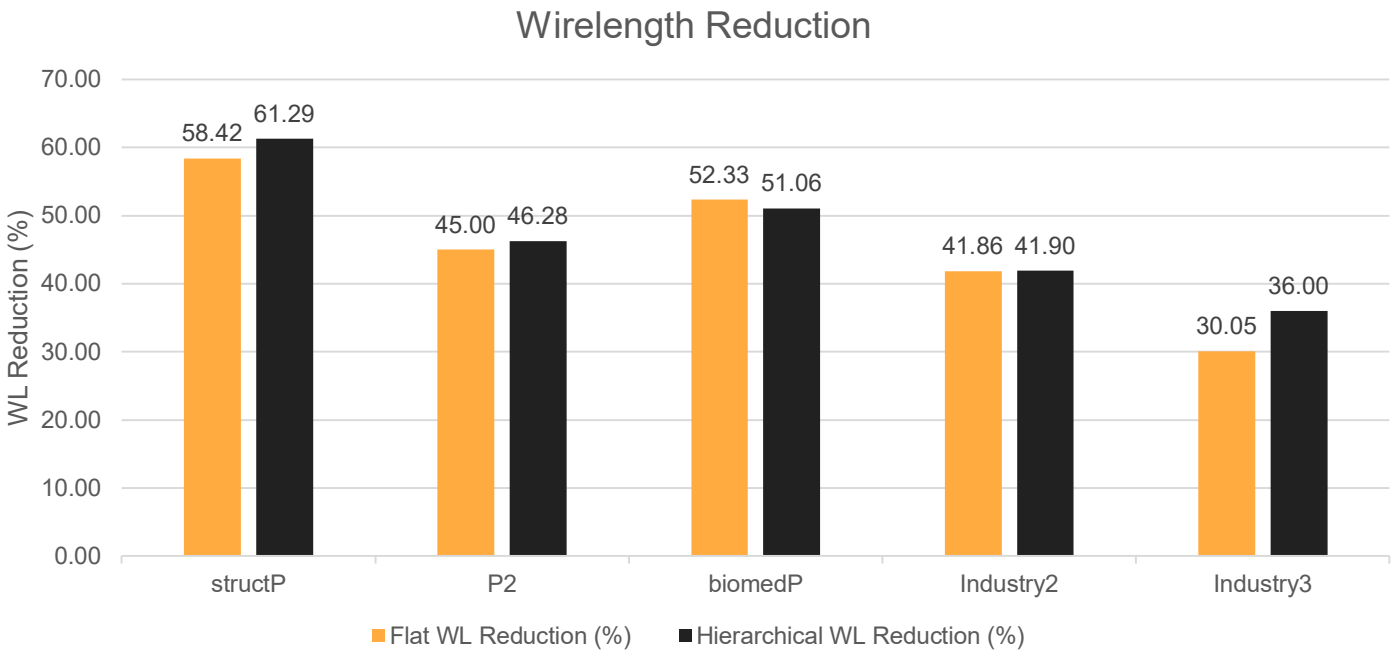


Wire length went down from 4,203,120.00 um to 2,689,910.00 um. Final rows' widths are balanced on average. There is no overlap anywhere and the IO cells are placed perfectly around the chip border.

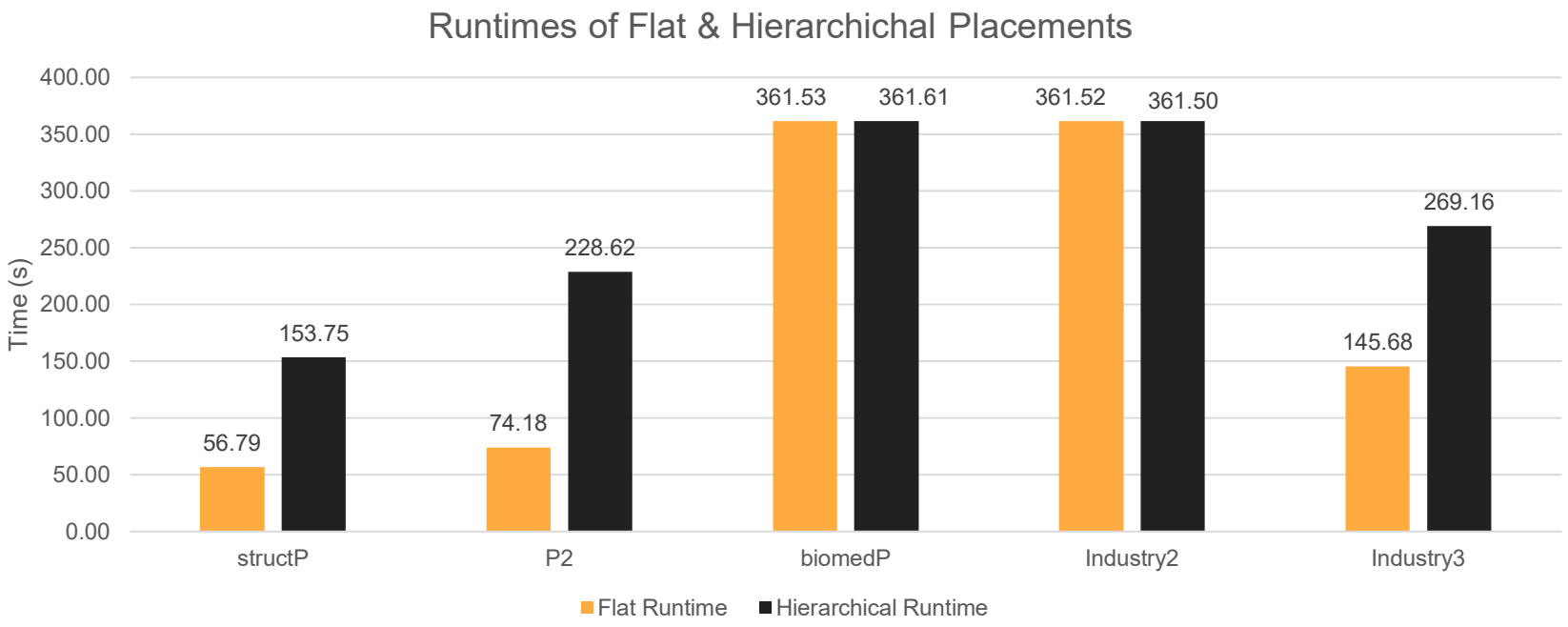
Wire Length Results Summary

	Initial WL	Final WL (Flat) (um)	WL Reduction (%)	Final WL (Hierarchical) (um)	WL Reduction (%)
structP	137,881.00	57,334.80	58.42	53,367.20	61.29
P2	326,839.00	179,760.00	45.00	175,578.00	46.28
biomedP	974,706.00	464,631.00	52.33	476,999.00	51.06
Industry2	2,646,960.00	1,539,060.00	41.86	1,538,010.00	41.90
Industry3	4,203,120.00	2,940,020.00	30.05	2,689,910.00	36.00

Wire Length Results Summary (continued)



Runtime Results Summary



Conclusion

- Flat and hierarchical implementations provide a significant wirelength reduction in a reasonable runtime.
- Hierarchical placement was shown to provide generally better wirelength reduction results than flat placement, at the cost of a longer runtime.
- Both of our implementations provide results with absolutely zero overlap.

References

[1] Sun, Wern-Jieh, and Carl Sechen. “Efficient and Effective Placement for Very Large Circuits.” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 14, no. 3, Mar. 1995, pp. 349–259.