



# Wire Length Prediction in Constraint Driven Placement

---

Qinghua Liu, Bo Hu, Malgorzata Marek-  
Sadowska

University of California, Santa Barbara



# Outline

---

- Motivation
- Constraint driven placement with wire length prediction
- Experimental results
- Conclusion and future work

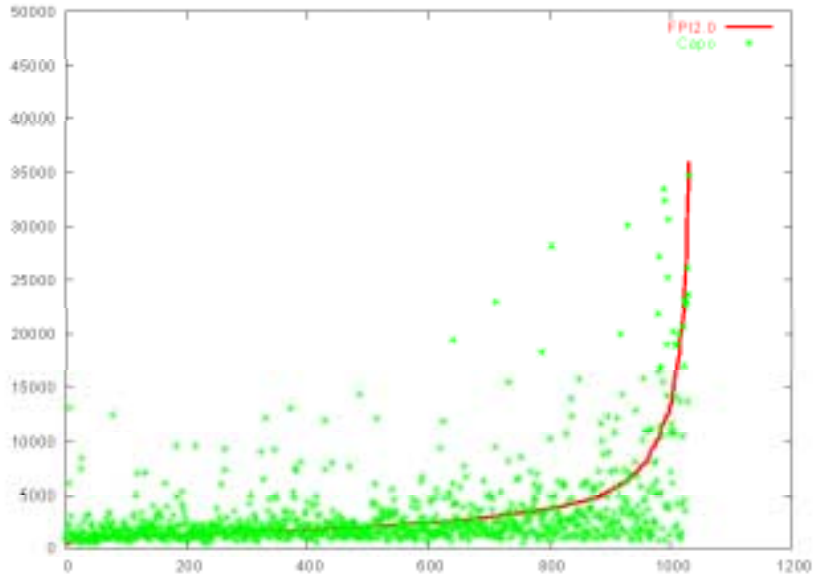


# Motivation

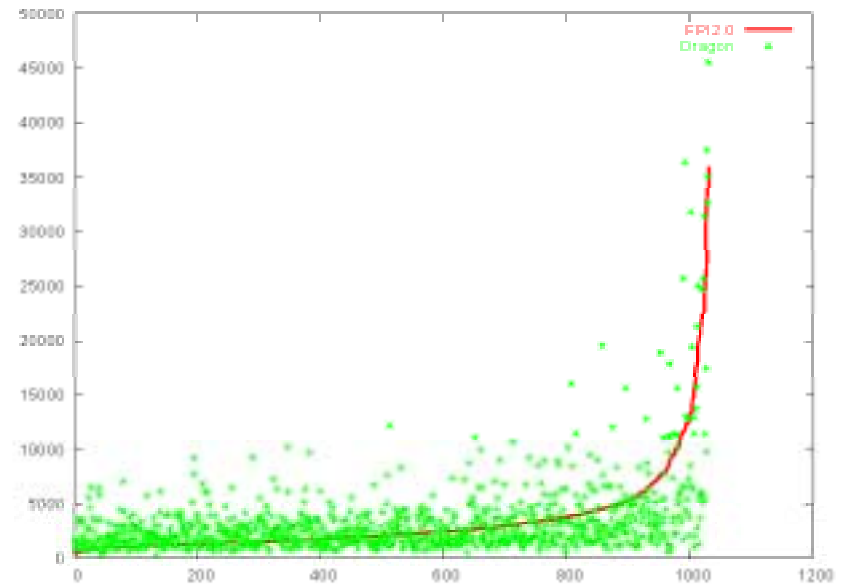
---

- Importance of individual wire length prediction
- Individual wire length is a function of the placement algorithm
- Constraints can help prediction

# Individual wire length VS Placer

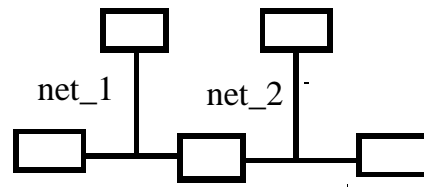


(a) FPI2.0 VS Capo

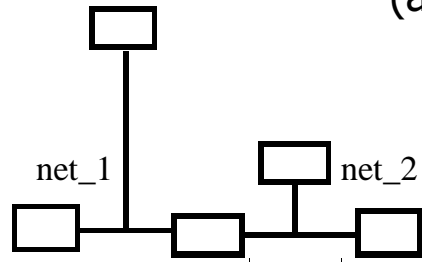


(b) FPI2.0 VS Dragon

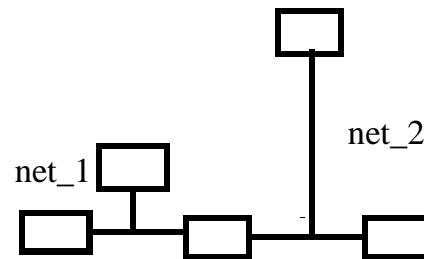
# Constraints help prediction



(a)



(b)



(c)



# Outline

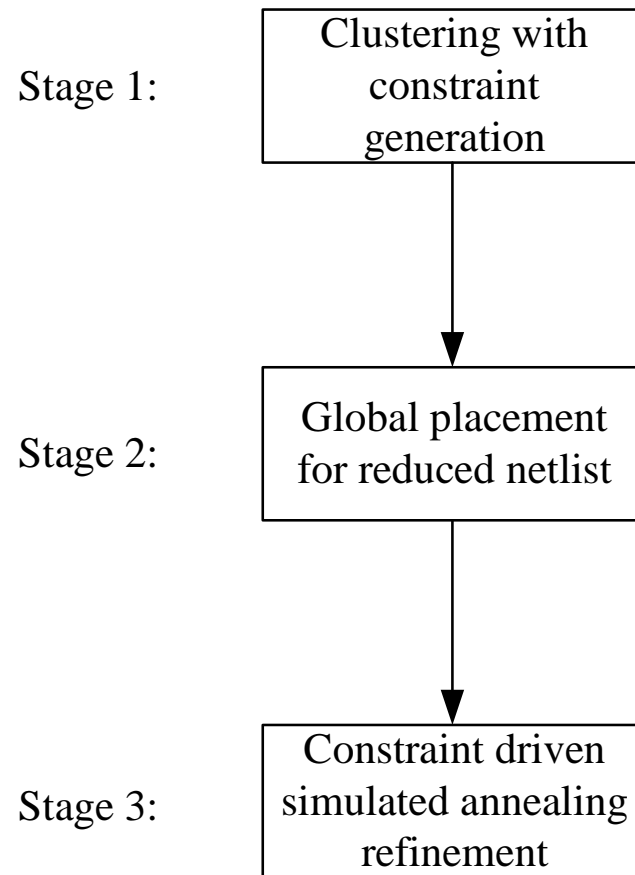
---

- Motivation
- Constraint driven placement with wire length prediction
- Experimental results
- Conclusion and future work



# Basic placement flow

---

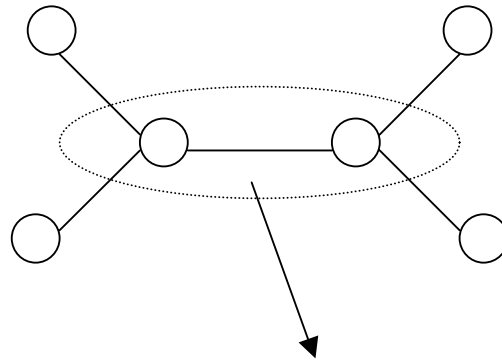


# Mutual contraction based clustering

- Mutual contraction

A metric to evaluate proximity of connected elements in a netlist

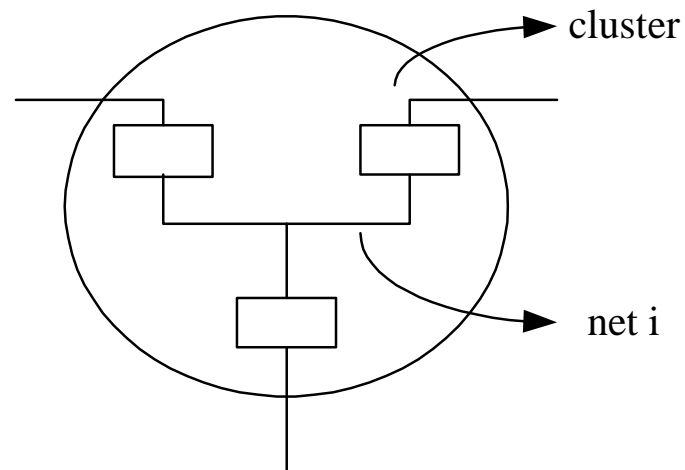
- Pairwise clustering strategy



Largest mutual contraction



# Wire length prediction



$$\text{Boundary}(i) = \frac{\sum W_j + \sum H_j}{2}$$



# Constraint driven placement

---

- Net-weighting
- Linear programming
- Simulated annealing

$$WL + \lambda \sum_{i=1}^{|E|} \max(L_i - c_i, 0)$$



# Outline

---

- Motivation
- Constraint driven placement with wire length prediction
- **Experimental results**
- Conclusion and future work

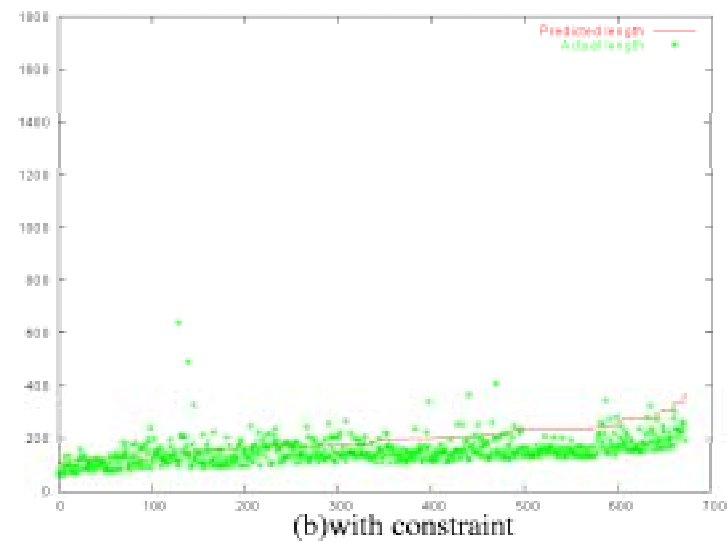
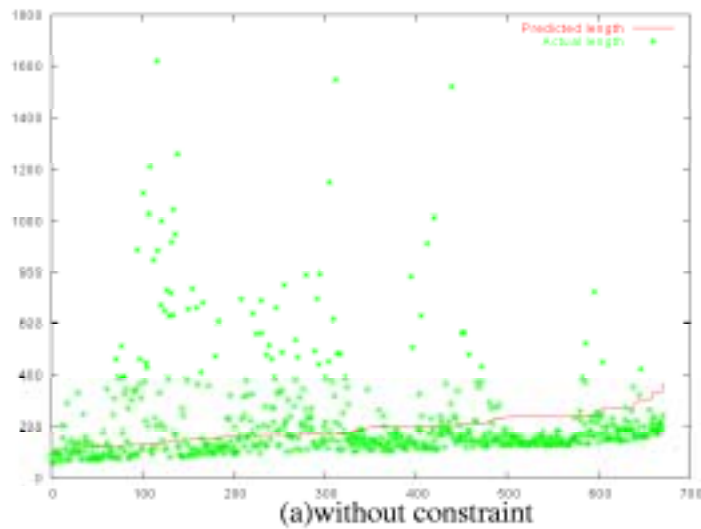


# Experimental results

---

- Constraint guided prediction
- Penalty for guidance of constraints
- Prediction VS Placement effort

# Constraint guided prediction



$$vio(i) = \frac{\max(l_i - c_p, 0)}{c_i}$$

ibm04	non-constraint	constraint
ave_vio	0.36	0.04
vio_sd	0.95	0.17



# Penalty for guidance of constraints

bench	#pre_nets	Capo twl	non-constraint			constraint		
			vio_len	twl	cpu	vio_len%	twl%	cpu%
ibm01	47.9%	55.8	1.94	54.4	155.65	0.17	1.01	1.07
ibm02	47.4%	159.8	3.44	157.9	324.12	0.26	0.99	1.04
ibm03	48.9%	10.1	0.277	9.59	328.45	0.10	1.00	1.03
ibm04	47.1%	13.1	0.632	12.5	393.11	0.09	1.01	1.06
ibm05	47.0%	35.0	0.995	34.3	488.39	0.10	0.99	1.05
ibm06	46.3%	14.9	0.373	15.4	590.92	0.13	0.98	1.05
ibm07	47.7%	370.0	10.92	358.7	697.04	0.14	1.01	1.05
ave			1.00	1.00	1.00	0.14	1.00	1.05



# Prediction VS Placement effort

bench	5		10		15	
	#pre_nets	cpu	#pre_nets	cpu%	#pre_nets	cpu%
ibm01	47.9%	166.99	59.9%	1.12	63.7%	1.27
ibm02	47.4%	338.00	56.4%	1.02	59.3%	1.15
ibm03	48.9%	337.74	63.6%	1.10	67.0%	1.21
ibm04	47.1%	417.61	61.7%	1.08	65.0%	1.15
ibm05	47.0%	513.86	62.0%	1.14	63.0%	1.21
ibm06	46.3%	620.9	59.7%	1.08	63.4%	1.15
ibm07	47.7%	734.04	60.6%	1.10	63.9%	1.21
ave	47.5%	1.00	60.6%	1.10	63.6%	1.18



# Conclusion

---

- Some knowledge of a specific placement flow is necessary for accurate wire length predictions
- Constraints, generated for the target placement flow, can be used to assist wire length predictions
- By enforcing constraints in placement, the length of interconnects can be predicted without placement quality losses





# Future work

---

- Multi-level clustering
- Delay budgeting with individual wire length prediction