Impact of FlowMap on my Research

Martin D.F. Wong
Hong Kong Baptist University

ISPD 2025

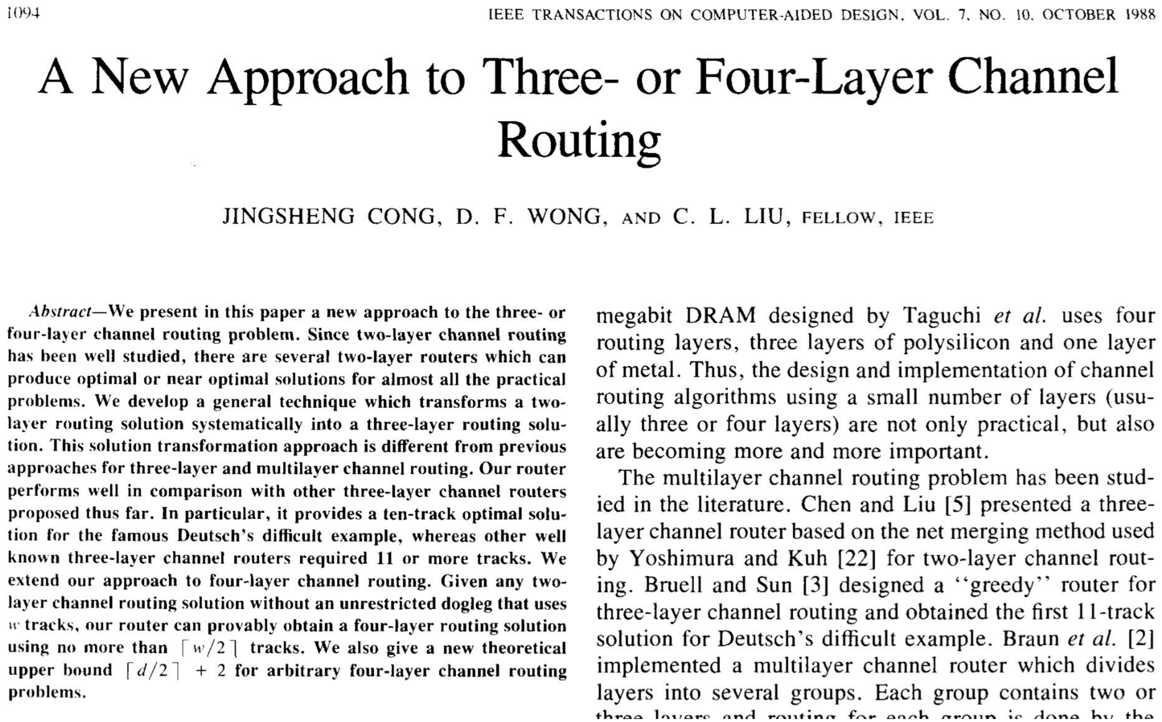
Our good old PhD student days

UIUC

Prof. C. L. Liu

First Research Collaboration

1st 10-track optimal solution of the Deutch's Difficult Problem!



1094 IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN, VOL. 7, NO. 10, OCTOBER 1988

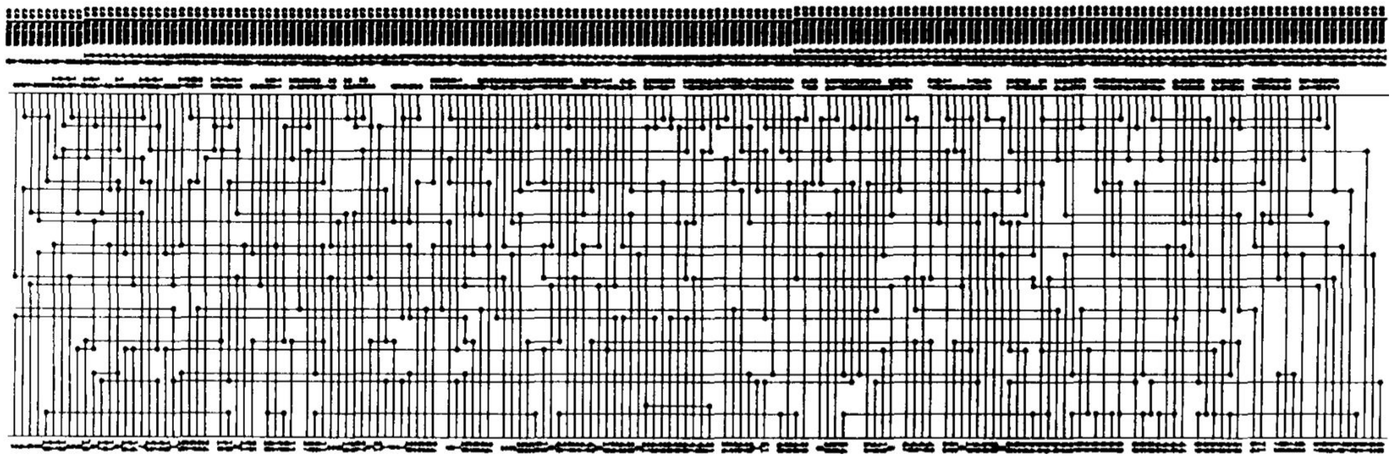
A New Approach to Three- or Four-Layer Channel Routing

JINGSHENG CONG, D. F. WONG, AND C. L. LIU, FELLOW, IEEE

Abstract—We present in this paper a new approach to the three- or four-layer channel routing problem. Since two-layer channel routing has been well studied, there are several two-layer routers which can produce optimal or near optimal solutions for almost all the practical problems. We develop a general technique which transforms a two-layer routing solution systematically into a three-layer routing solution. This solution transformation approach is different from previous approaches for three-layer and multilayer channel routing. Our router performs well in comparison with other three-layer channel routers proposed thus far. In particular, it provides a ten-track optimal solution for the famous Deutsch's difficult example, whereas other well known three-layer channel routers required 11 or more tracks. We extend our approach to four-layer channel routing. Given any two-layer channel routing solution without an unrestricted dogleg that uses w tracks, our router can provably obtain a four-layer routing solution using no more than $\lceil w/2 \rceil$ tracks. We also give a new theoretical upper bound $\lceil d/2 \rceil + 2$ for arbitrary four-layer channel routing problems.

megabit DRAM designed by Taguchi *et al.* uses four routing layers, three layers of polysilicon and one layer of metal. Thus, the design and implementation of channel routing algorithms using a small number of layers (usually three or four layers) are not only practical, but also are becoming more and more important.

The multilayer channel routing problem has been studied in the literature. Chen and Liu [5] presented a three-layer channel router based on the net merging method used by Yoshimura and Kuh [22] for two-layer channel routing. Bruell and Sun [3] designed a "greedy" router for three-layer channel routing and obtained the first 11-track solution for Deutsch's difficult example. Braun *et al.* [2] implemented a multilayer channel router which divides layers into several groups. Each group contains two or three layers and routing for each group is done by the

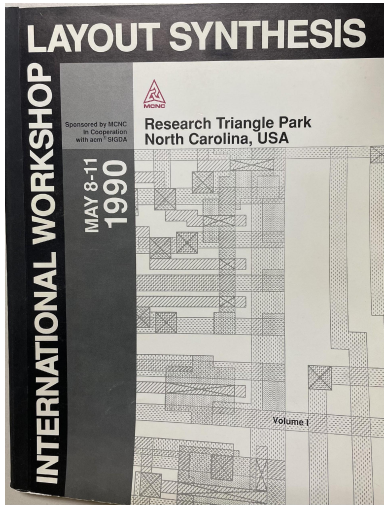


The Most Significant Collaboration with Jason

Jason was with ISPD right from the start

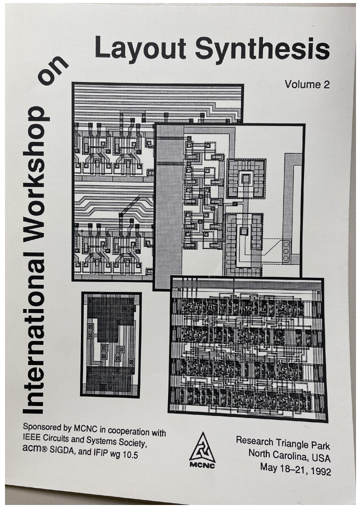
History of ISPD

- ACM SIGDA Physical Design Workshop: 1987, 1989, 1991, 1993, 1996
- MCNC Layout Synthesis Workshop: 1988, 1990, 1992
- ISPD: 1997-Present



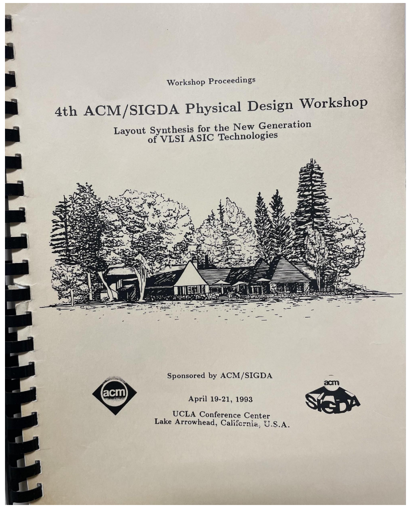
LAYOUT SYNTHESIS
INTERNATIONAL WORKSHOP
MAY 8-11
1990
Sponsored by MCNC
In Cooperation
with acm® SIGDA
Research Triangle Park
North Carolina, USA
Volume I

1990



International Workshop on
Layout Synthesis
Volume 2
Sponsored by MCNC in cooperation with
IEEE Circuits and Systems Society,
acm® SIGDA, and IFIP wg 10.5
Research Triangle Park
North Carolina, USA
May 18–21, 1992

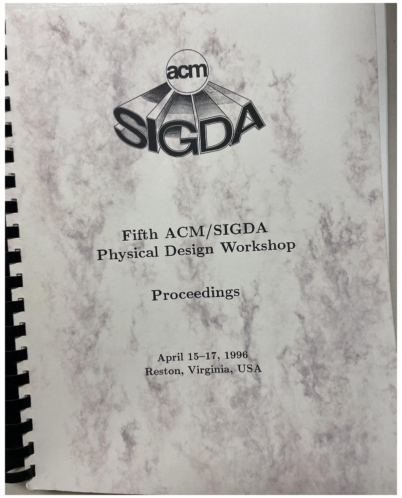
1992



Workshop Proceedings
4th ACM/SIGDA Physical Design Workshop
Layout Synthesis for the New Generation
of VLSI ASIC Technologies
Sponsored by ACM/SIGDA
April 19-21, 1993
UCLA Conference Center
Lake Arrowhead, California, U.S.A.

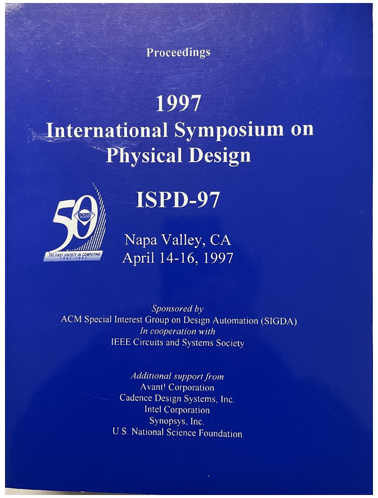
1993

UCLA



Fifth ACM/SIGDA
Physical Design Workshop
Proceedings
April 15–17, 1996
Reston, Virginia, USA

1996



Proceedings
1997
International Symposium on
Physical Design
ISPD-97
Napa Valley, CA
April 14-16, 1997
Sponsored by
ACM Special Interest Group on Design Automation (SIGDA)
In cooperation with
IEEE Circuits and Systems Society
Additional support from
Avant! Corporation
Cadence Design Systems, Inc.
Intel Corporation
Synopsys, Inc.
U.S. National Science Foundation

1997

FlowMap

ICCAD 1992, TCAD Jan 1994

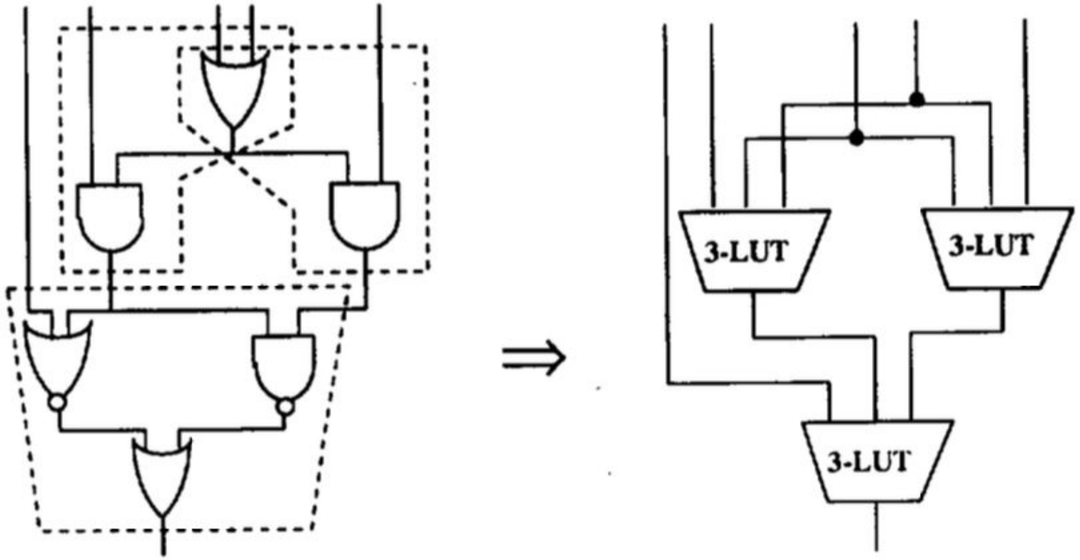
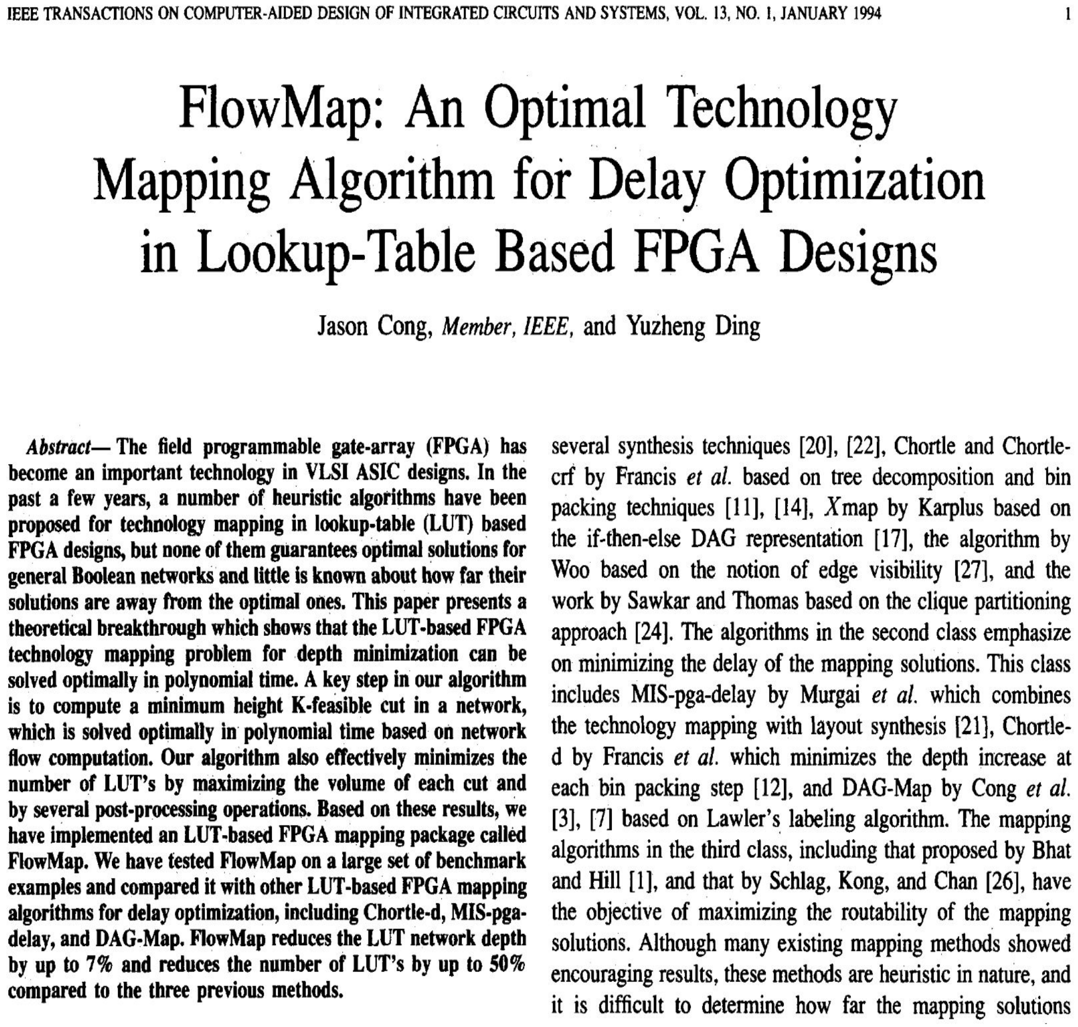


Fig. 1. Mapping a Boolean network to a K-LUT network ($K = 3$).

- FPGA (LUT-based) Technology Mapping
- Allow gate replications
- Novel application of Network Flow
- Polynomial time optimal algorithm
- Guarantees minimum delay


IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 13, NO. 1, JANUARY 1994 1

FlowMap: An Optimal Technology Mapping Algorithm for Delay Optimization in Lookup-Table Based FPGA Designs

Jason Cong, *Member, IEEE*, and Yuzheng Ding

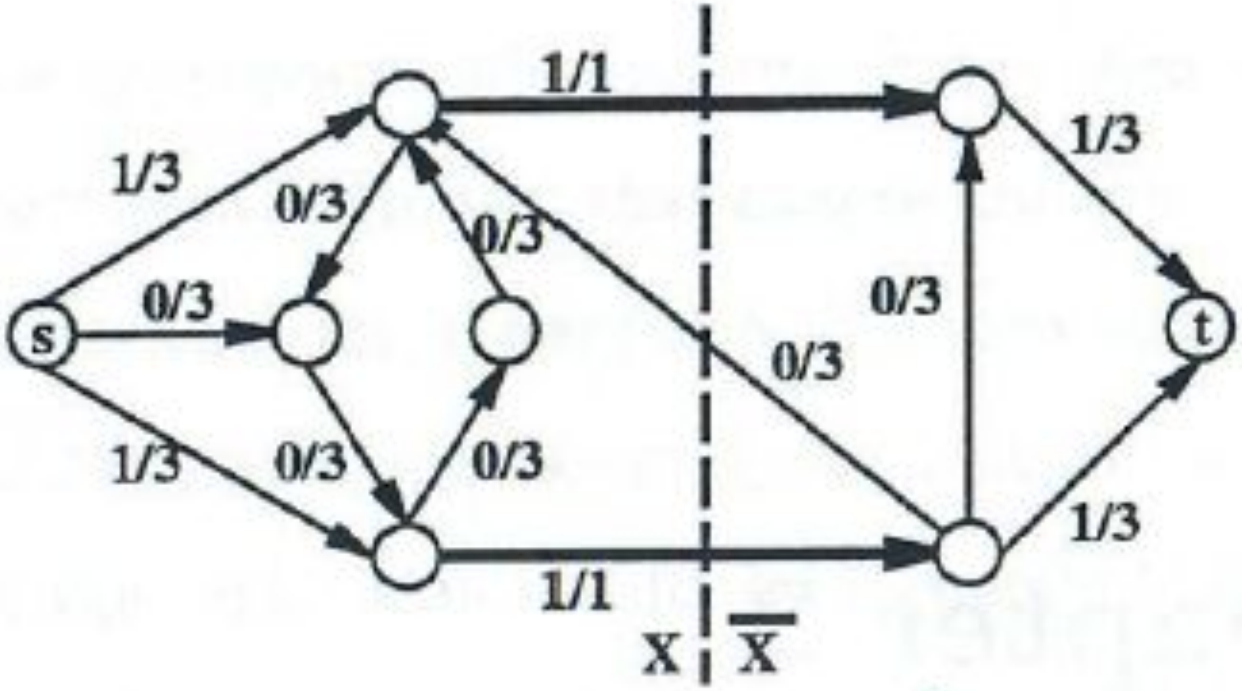
***Abstract*— The field programmable gate-array (FPGA) has become an important technology in VLSI ASIC designs. In the past a few years, a number of heuristic algorithms have been proposed for technology mapping in lookup-table (LUT) based FPGA designs, but none of them guarantees optimal solutions for general Boolean networks and little is known about how far their solutions are away from the optimal ones. This paper presents a theoretical breakthrough which shows that the LUT-based FPGA technology mapping problem for depth minimization can be solved optimally in polynomial time. A key step in our algorithm is to compute a minimum height K-feasible cut in a network, which is solved optimally in polynomial time based on network flow computation. Our algorithm also effectively minimizes the number of LUT's by maximizing the volume of each cut and by several post-processing operations. Based on these results, we have implemented an LUT-based FPGA mapping package called FlowMap. We have tested FlowMap on a large set of benchmark examples and compared it with other LUT-based FPGA mapping algorithms for delay optimization, including Chortle-d, MIS-pga-delay, and DAG-Map. FlowMap reduces the LUT network depth by up to 7% and reduces the number of LUT's by up to 50% compared to the three previous methods.**

several synthesis techniques [20], [22], Chortle and Chortle-crf by Francis *et al.* based on tree decomposition and bin packing techniques [11], [14], *X*map by Karplus based on the if-then-else DAG representation [17], the algorithm by Woo based on the notion of edge visibility [27], and the work by Sawkar and Thomas based on the clique partitioning approach [24]. The algorithms in the second class emphasize on minimizing the delay of the mapping solutions. This class includes MIS-pga-delay by Murgai *et al.* which combines the technology mapping with layout synthesis [21], Chortle-d by Francis *et al.* which minimizes the depth increase at each bin packing step [12], and DAG-Map by Cong *et al.* [3], [7] based on Lawler's labeling algorithm. The mapping algorithms in the third class, including that proposed by Bhat and Hill [1], and that by Schlag, Kong, and Chan [26], have the objective of maximizing the routability of the mapping solutions. Although many existing mapping methods showed encouraging results, these methods are heuristic in nature, and it is difficult to determine how far the mapping solutions

Network Flow

Max Flow = Min Cut

Capacity of a Cut = Sum of the Capacities of all Forward Cut Edges



flow/capacity

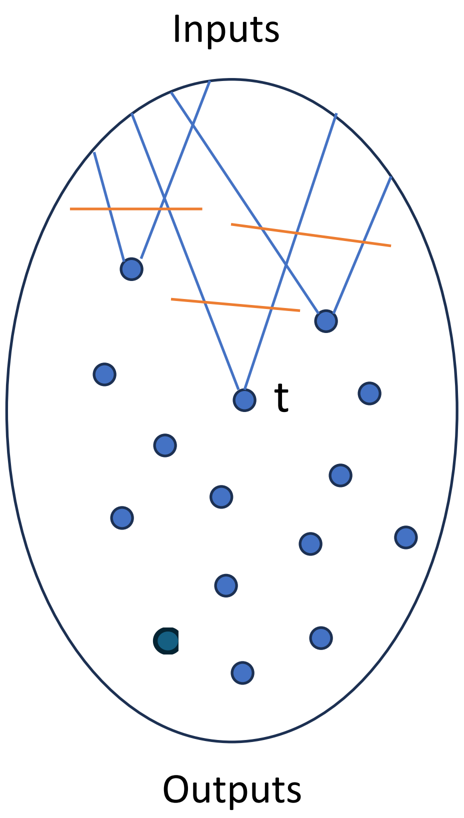
max–flow $|f| = 1 + 1 = 2$

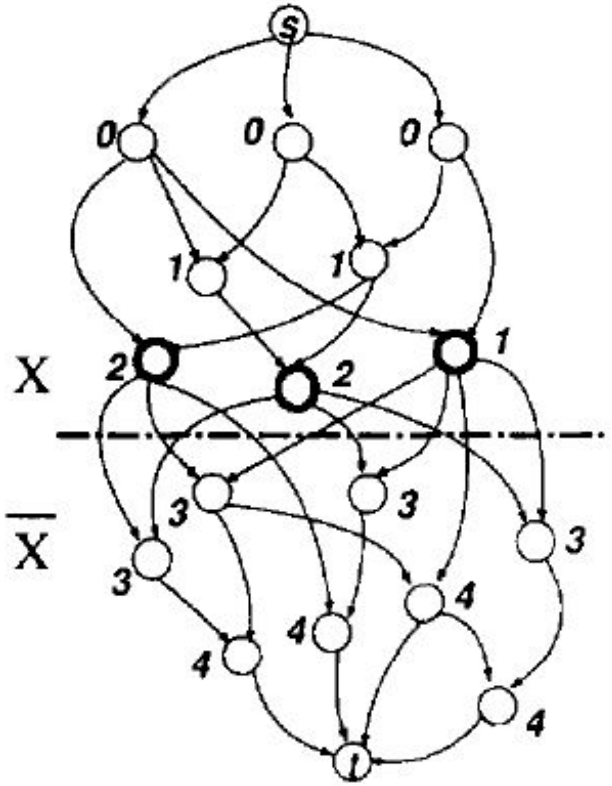
$\text{capacity}(X, \overline{X}) = 1 + 1 = 2$

The FlowMap Algorithm

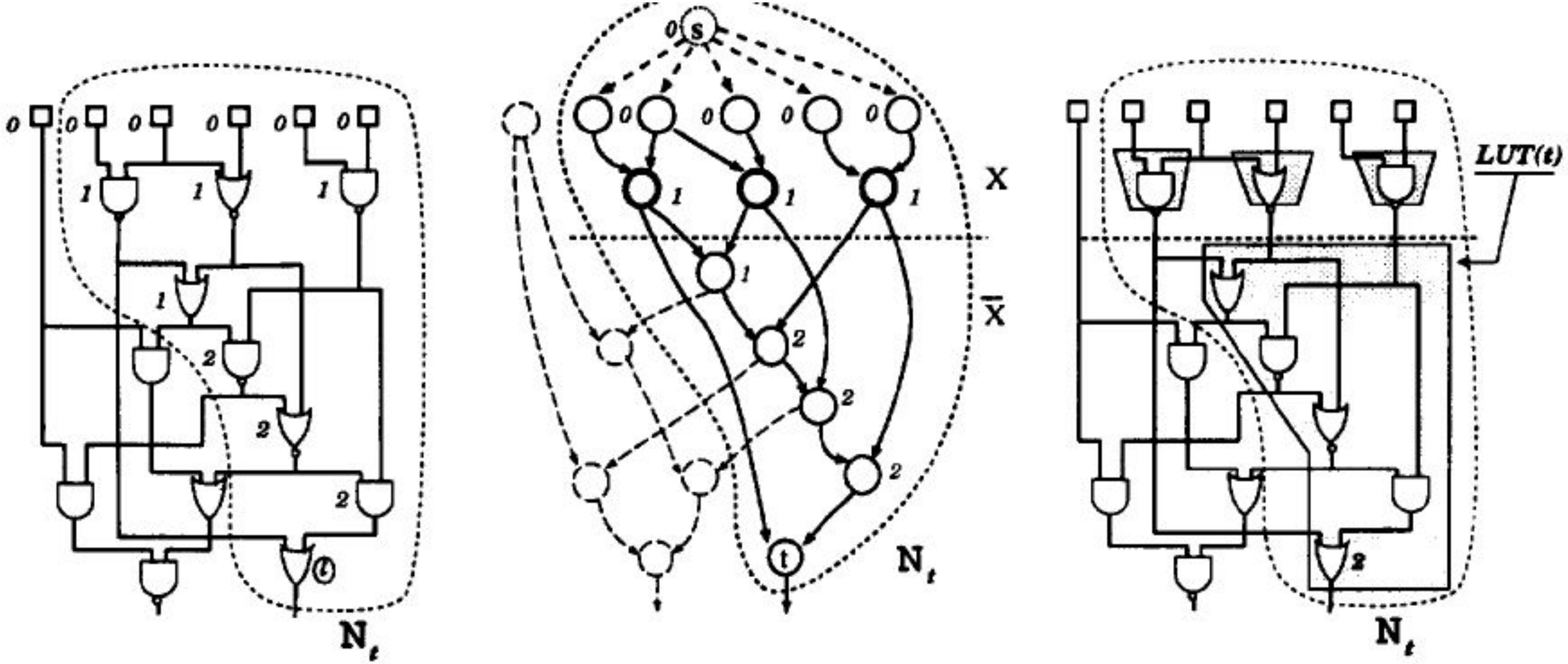
- Top-Down <u>Labeling Phase</u>
- Bottom-Up <u>Mapping Phase</u>

Dynamic Programming!





A 3-feasible cut of edge cut-size 10, volume 9, and height 2.



(a) (b) (c)

Computing the label $l(t)$ of node t ($K = 3$). (a) The partial network. (b) Construction of N_t and the highest 3-feasible cut. (c) Determining $l(t)$.

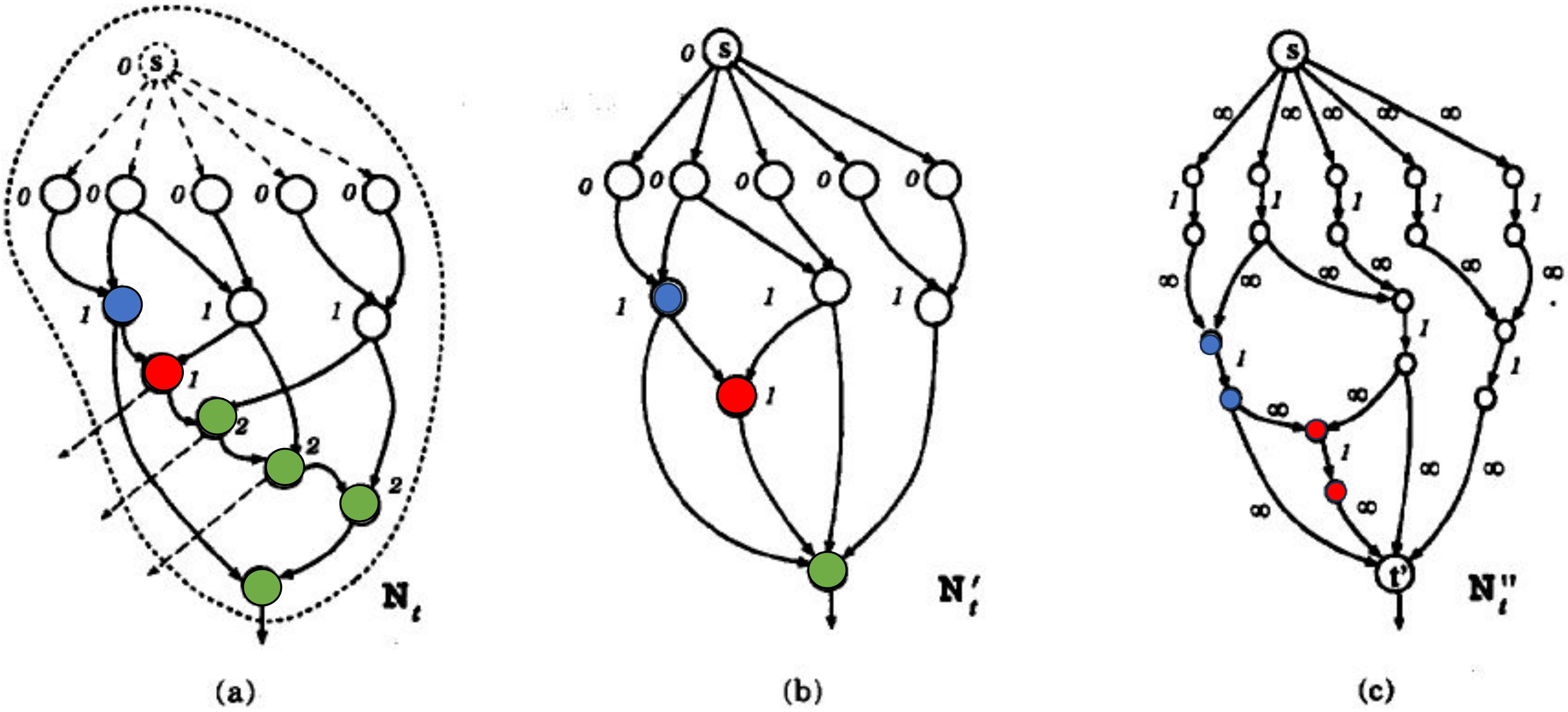
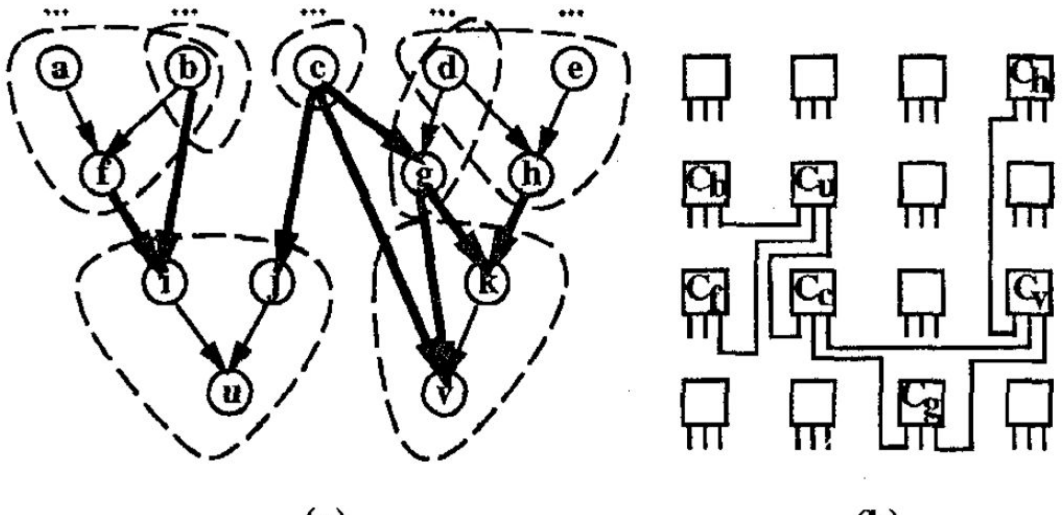


Fig. 5. Network transformations in computing a minimum height K-feasible cut in N_t ($K = 3$).

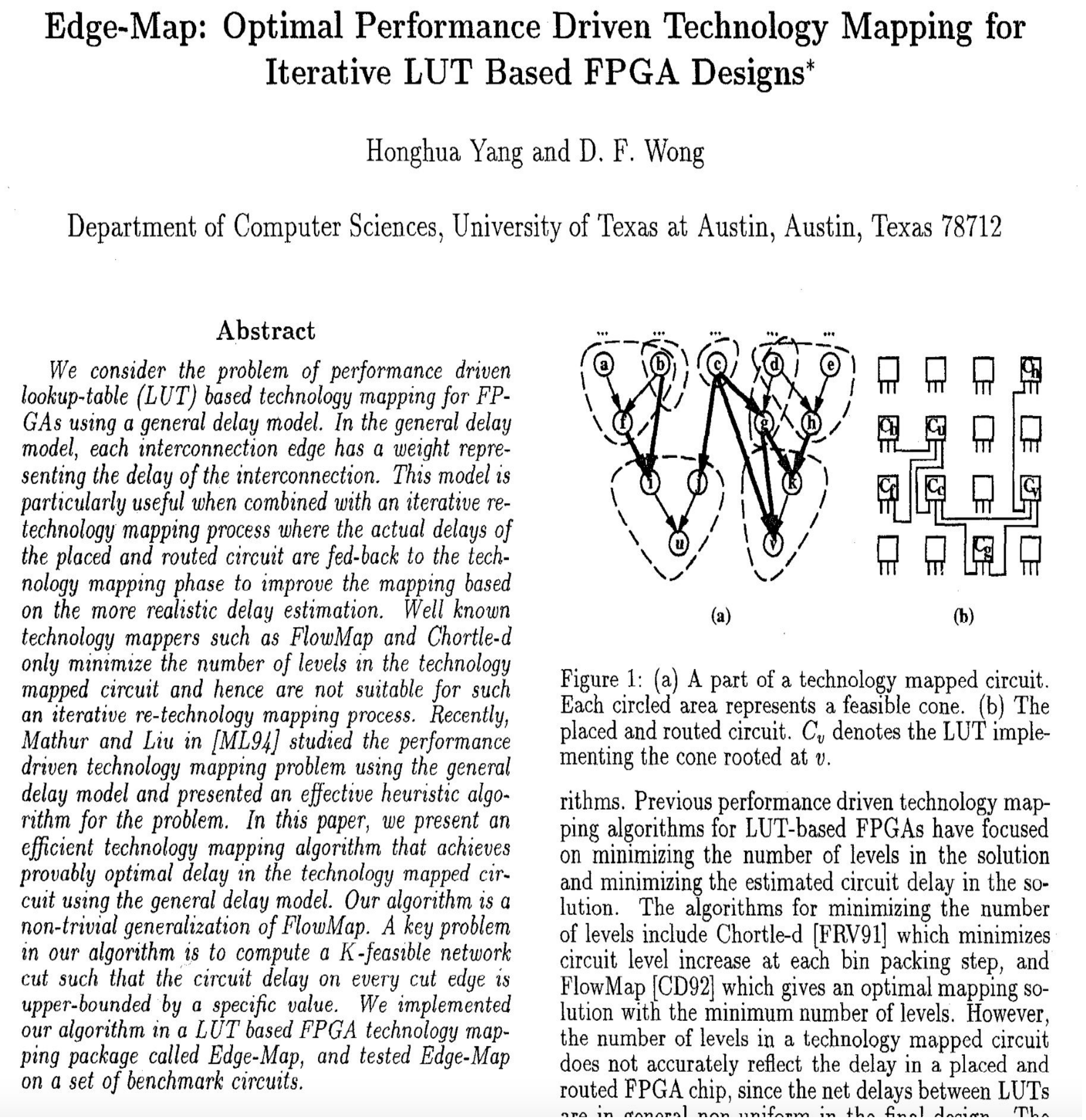
Edge-Map



(a) (b)

- Each edge has an estimated delay value
- Only delay on "visible" edges, i.e. edges connecting LUTs, are counted

ICCAD 1994



Edge-Map: Optimal Performance Driven Technology Mapping for Iterative LUT Based FPGA Designs*

Honghua Yang and D. F. Wong

Department of Computer Sciences, University of Texas at Austin, Austin, Texas 78712

Abstract

We consider the problem of performance driven lookup-table (LUT) based technology mapping for FPGAs using a general delay model. In the general delay model, each interconnection edge has a weight representing the delay of the interconnection. This model is particularly useful when combined with an iterative re-technology mapping process where the actual delays of the placed and routed circuit are fed-back to the technology mapping phase to improve the mapping based on the more realistic delay estimation. Well known technology mappers such as FlowMap and Chortle-d only minimize the number of levels in the technology mapped circuit and hence are not suitable for such an iterative re-technology mapping process. Recently, Mathur and Liu in [ML94] studied the performance driven technology mapping problem using the general delay model and presented an effective heuristic algorithm for the problem. In this paper, we present an efficient technology mapping algorithm that achieves provably optimal delay in the technology mapped circuit using the general delay model. Our algorithm is a non-trivial generalization of FlowMap. A key problem in our algorithm is to compute a K-feasible network cut such that the circuit delay on every cut edge is upper-bounded by a specific value. We implemented our algorithm in a LUT based FPGA technology mapping package called Edge-Map, and tested Edge-Map on a set of benchmark circuits.

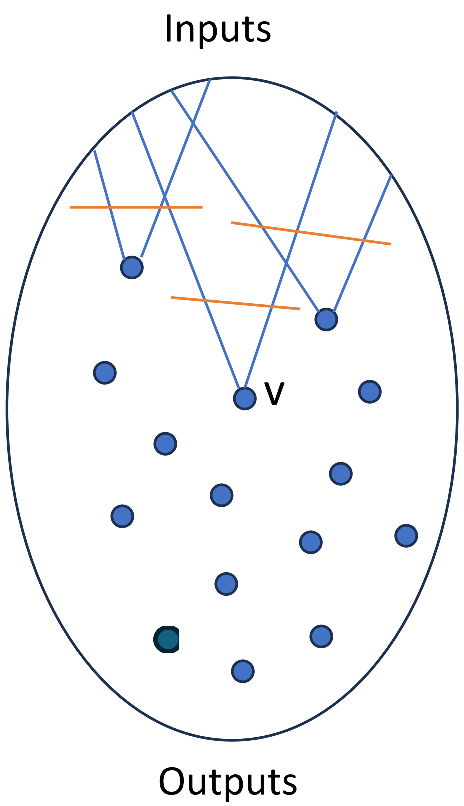
(a) (b)

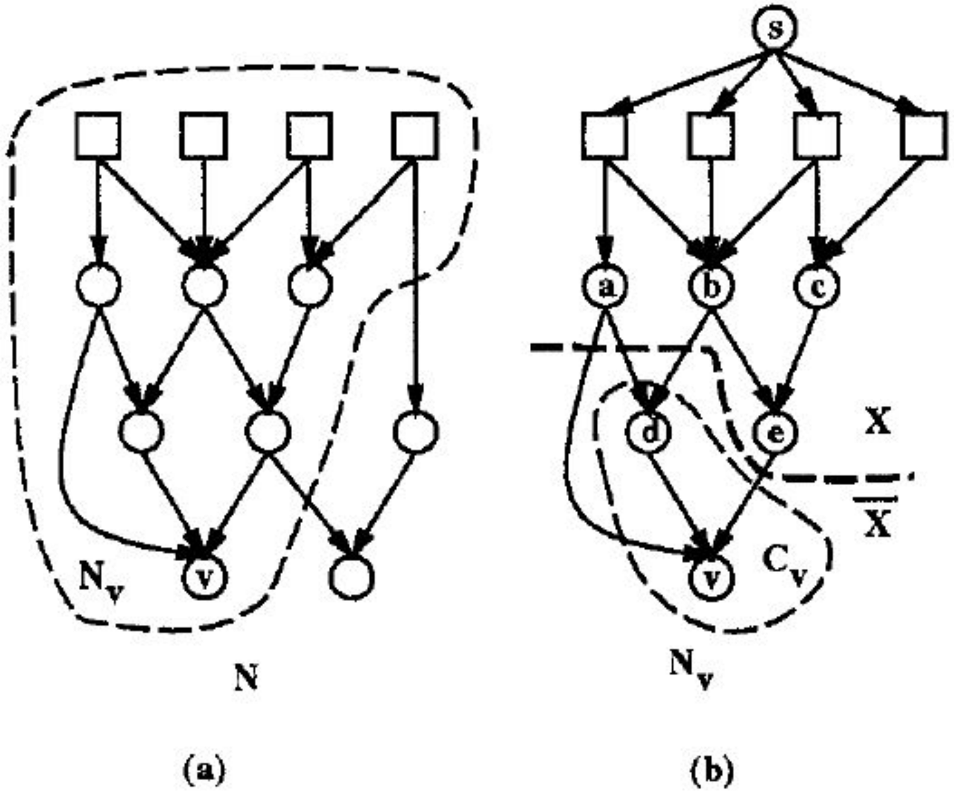
Figure 1: (a) A part of a technology mapped circuit. Each circled area represents a feasible cone. (b) The placed and routed circuit. C_v denotes the LUT implementing the cone rooted at v.

rithms. Previous performance driven technology mapping algorithms for LUT-based FPGAs have focused on minimizing the number of levels in the solution and minimizing the estimated circuit delay in the solution. The algorithms for minimizing the number of levels include Chortle-d [FRV91] which minimizes circuit level increase at each bin packing step, and FlowMap [CD92] which gives an optimal mapping solution with the minimum number of levels. However, the number of levels in a technology mapped circuit does not accurately reflect the delay in a placed and routed FPGA chip, since the net delays between LUTs

Edge-Map Algorithm

- Top-Down <u>Labeling Phase</u>
- Bottom-Up <u>Mapping Phase</u>
- Dynamic Programming!

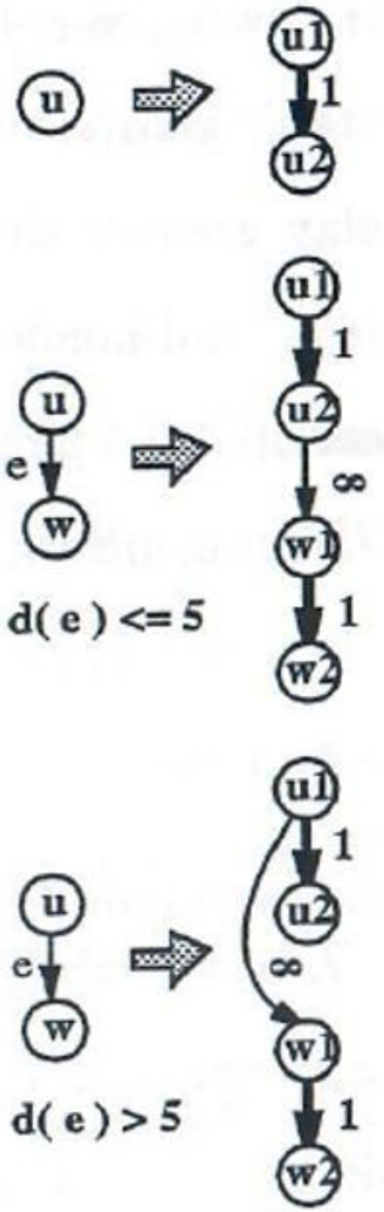


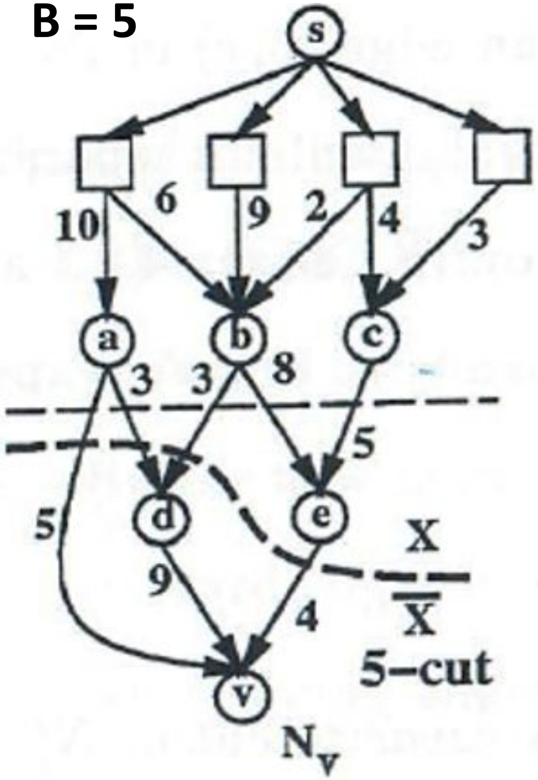


Is there a B-Cut?

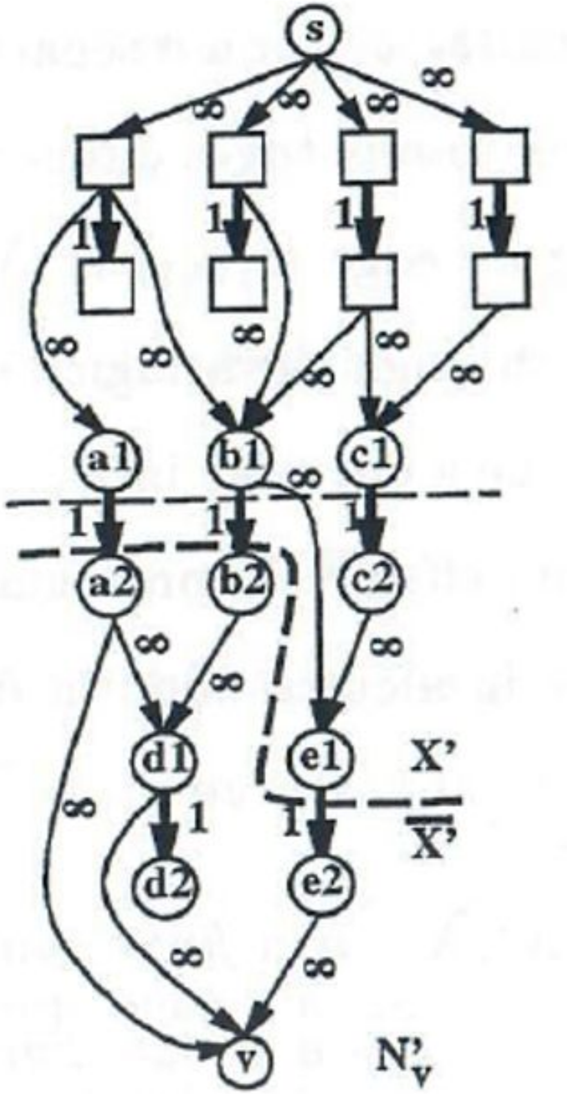
B = 2, 3, 4, 5, 6, 9, 10

B = 5





(a)
Labels are d(e) values.



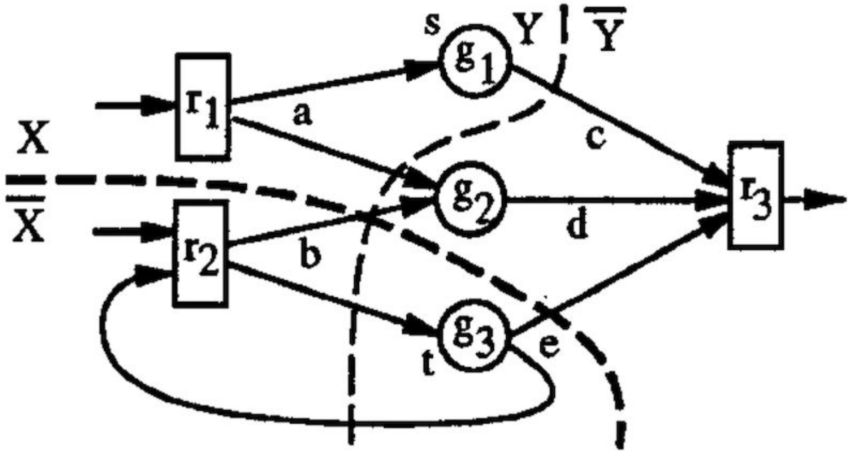
(b)
Labels are flow capacities.

FBB

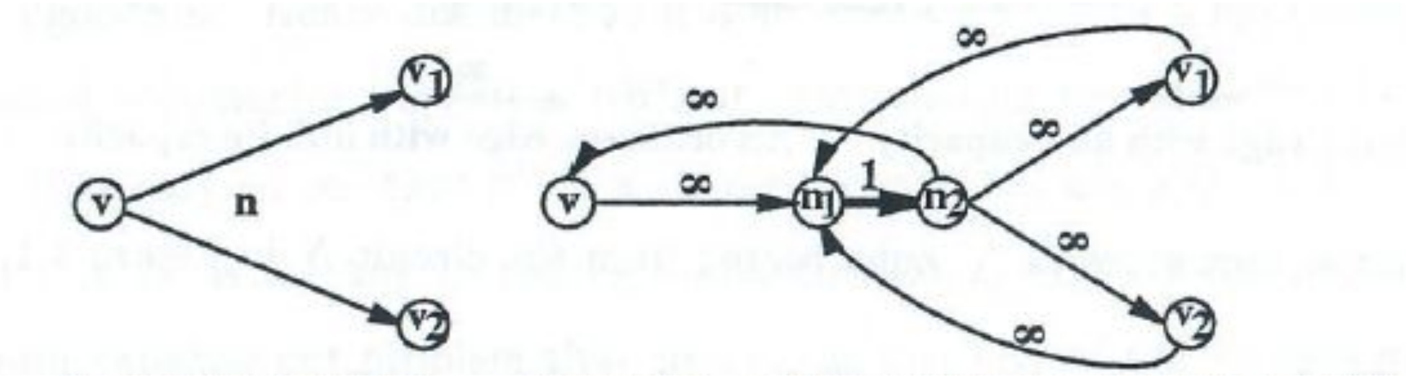
- Yang and Wong (ICCAD-1994, TCAD)
- Finding a **min net cut** in a circuit based on network flow
- Iterative application of the min-cut algorithm to get a **balanced** min-net-cut

Cut Size = 3

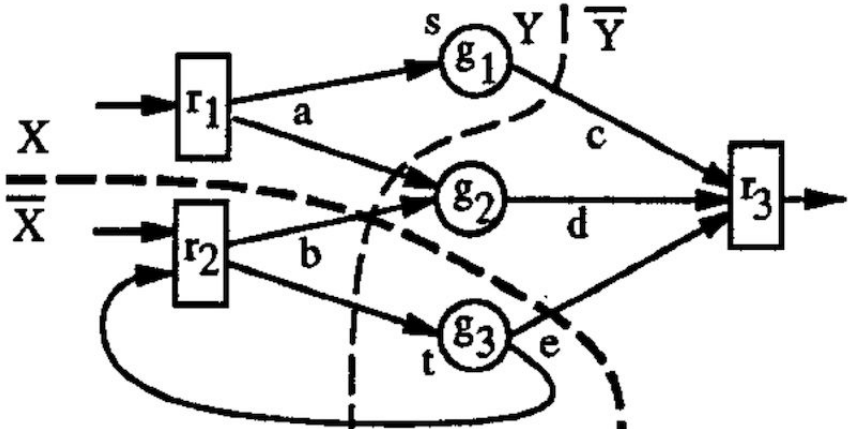
2



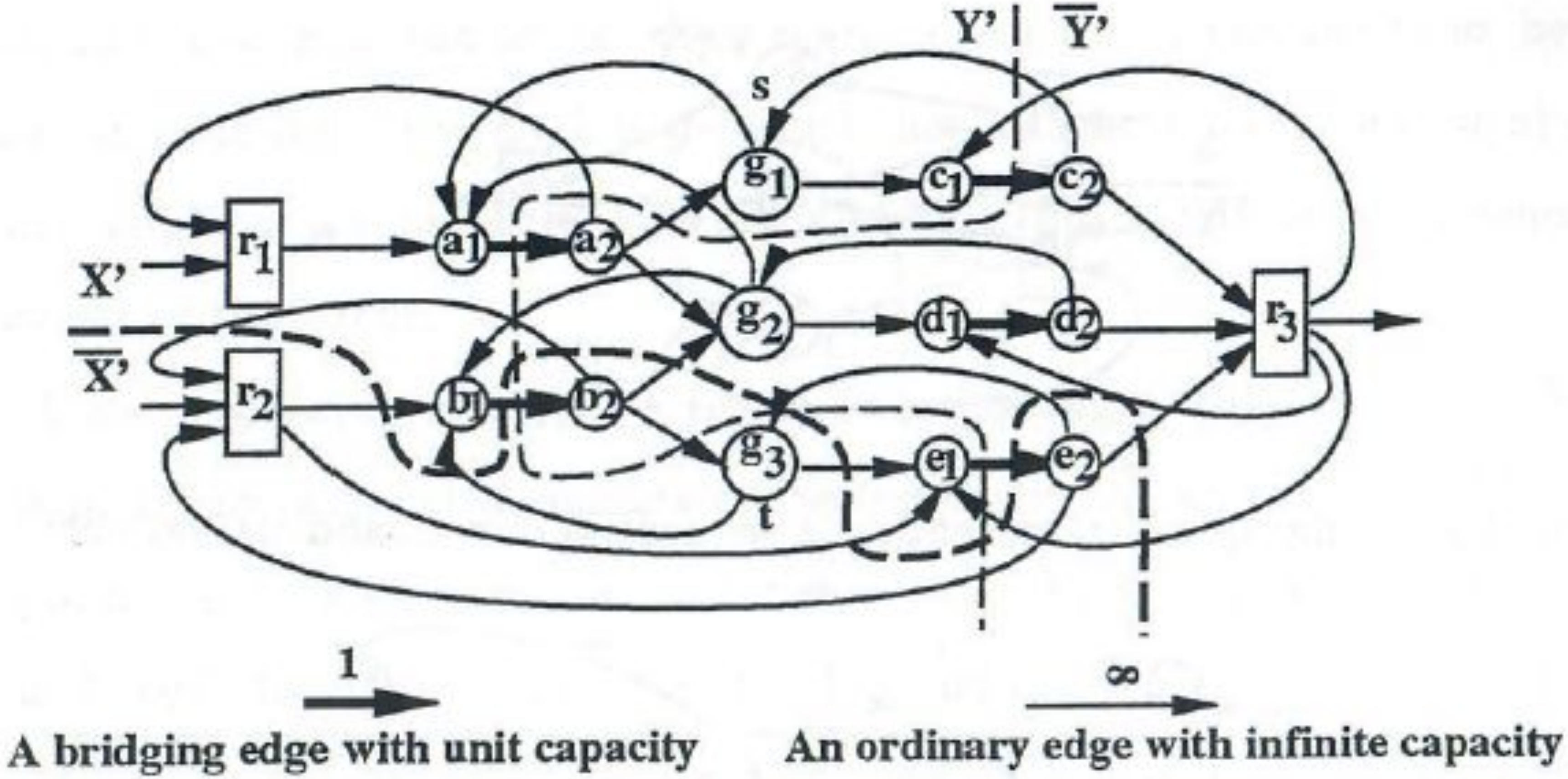
Finding a Min Cut



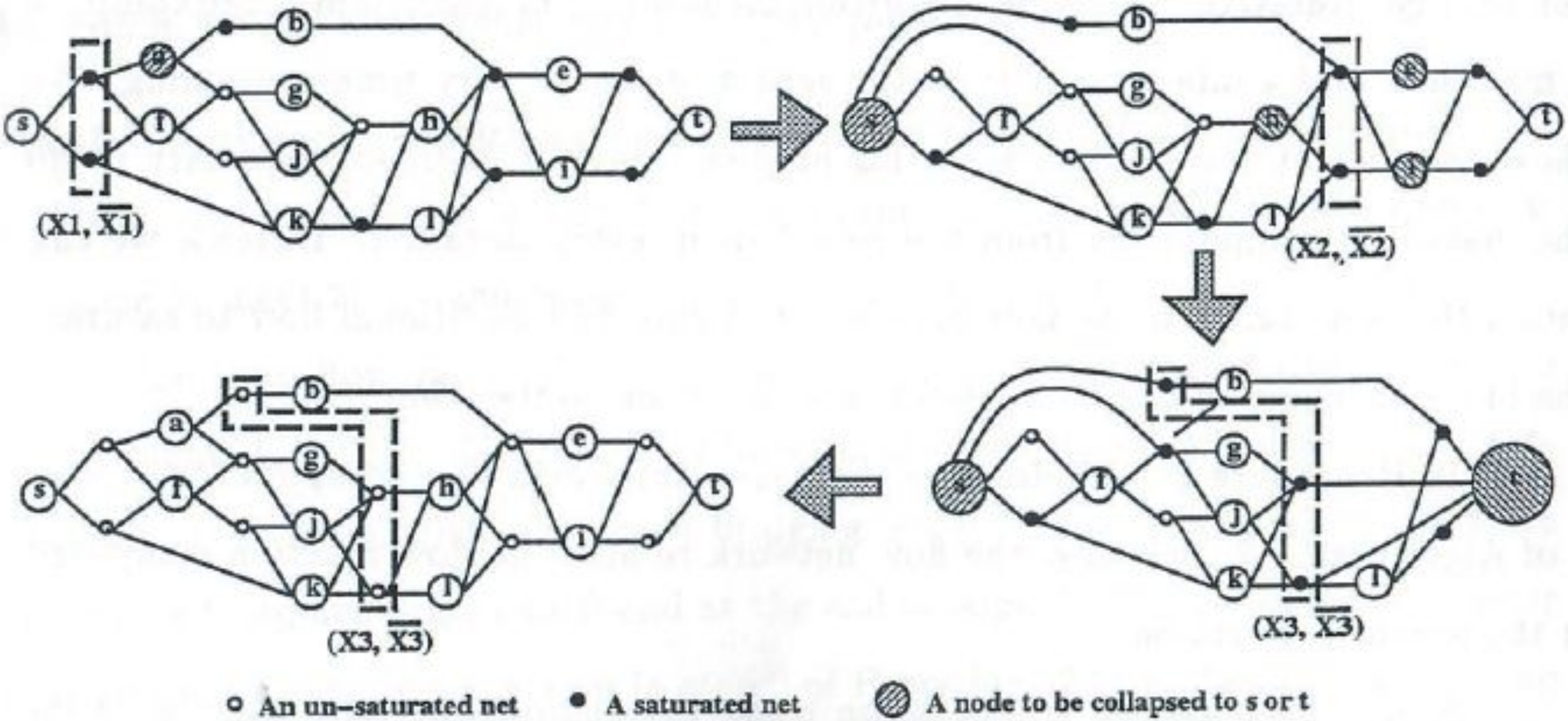
Modelling a net in a flow network



s
Y
$\overline{Y}$
X
$\overline{X}$
r_1
r_2
r_3
g_1
g_2
g_3
a
b
c
d
e
t

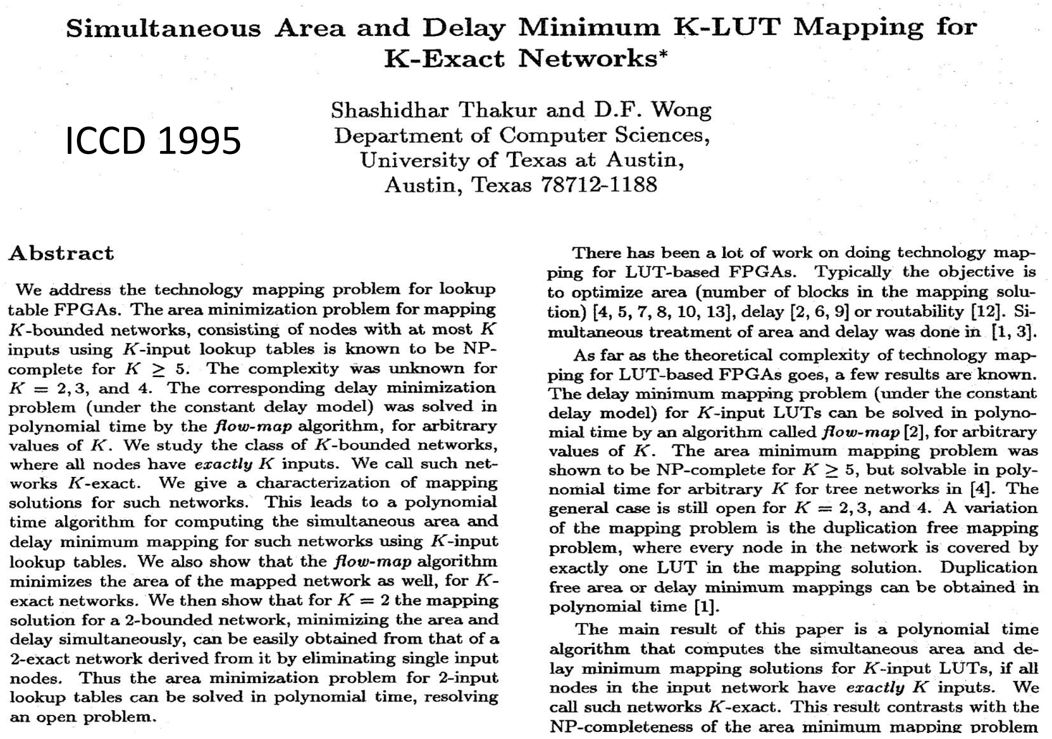


Y'
$\overline{Y}$'
s
X'
$\overline{X}$'
r_1
r_2
r_3
a_1
a_2
b_1
b_2
g_1
g_2
g_3
c_1
c_2
d_1
d_2
e_1
e_2
t
1
∞
A bridging edge with unit capacity
An ordinary edge with infinite capacity


s
a
b
f
g
j
k
h
l
e
i
t
(X1, $\overline{X1}$)
(X2, $\overline{X2}$)
(X3, $\overline{X3}$)
(X3, $\overline{X3}$)
○ An un-saturated net
● A saturated net
▨ A node to be collapsed to s or t

Simultaneous Area and Delay Minimization

- FPGA K-LUT Technology Mapping
- K-Bounded Networks
  - Delay minimization (FlowMap)
  - Area minimization (NP-Complete for K >= 5)
  - Area minimization is open for K = 2, 3, 4
- K-Exact Networks
  - FlowMap Algorithm also minimizes area
  - Area & delay minimization solved in polynomial time
- 2-Bounded Networks
  - Remove single-input nodes □ 2-exact networks
  - Area & delay minimization solved in polynomial time
  - Solved the open problem of area minimization for K=2



ICCD 1995

Simultaneous Area and Delay Minimum K-LUT Mapping for K-Exact Networks*

Shashidhar Thakur and D.F. Wong
Department of Computer Sciences,
University of Texas at Austin,
Austin, Texas 78712-1188

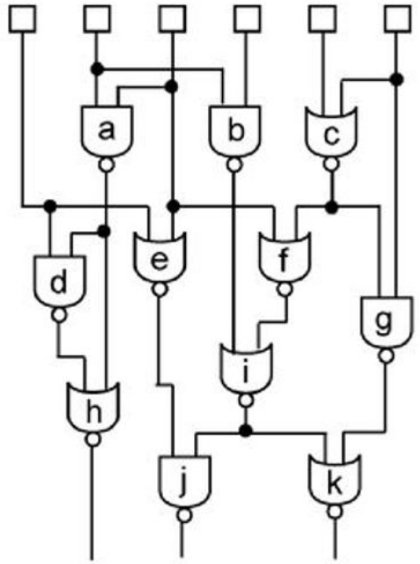
Abstract

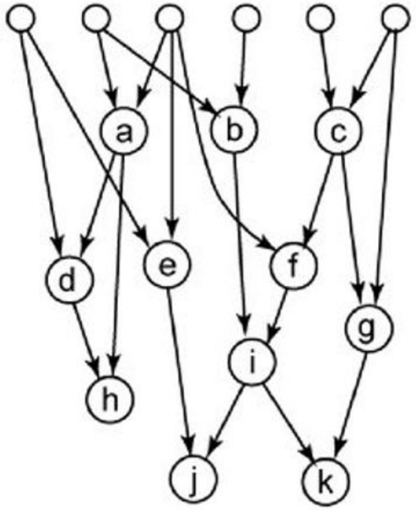
We address the technology mapping problem for lookup table FPGAs. The area minimization problem for mapping K-bounded networks, consisting of nodes with at most K inputs using K-input lookup tables is known to be NP-complete for $K \geq 5$. The complexity was unknown for $K = 2, 3$, and 4. The corresponding delay minimization problem (under the constant delay model) was solved in polynomial time by the *flow-map* algorithm, for arbitrary values of K. We study the class of K-bounded networks, where all nodes have *exactly* K inputs. We call such networks K-exact. We give a characterization of mapping solutions for such networks. This leads to a polynomial time algorithm for computing the simultaneous area and delay minimum mapping for such networks using K-input lookup tables. We also show that the *flow-map* algorithm minimizes the area of the mapped network as well, for K-exact networks. We then show that for $K = 2$ the mapping solution for a 2-bounded network, minimizing the area and delay simultaneously, can be easily obtained from that of a 2-exact network derived from it by eliminating single input nodes. Thus the area minimization problem for 2-input lookup tables can be solved in polynomial time, resolving an open problem.

There has been a lot of work on doing technology mapping for LUT-based FPGAs. Typically the objective is to optimize area (number of blocks in the mapping solution) [4, 5, 7, 8, 10, 13], delay [2, 6, 9] or routability [12]. Simultaneous treatment of area and delay was done in [1, 3].

As far as the theoretical complexity of technology mapping for LUT-based FPGAs goes, a few results are known. The delay minimum mapping problem (under the constant delay model) for K-input LUTs can be solved in polynomial time by an algorithm called *flow-map* [2], for arbitrary values of K. The area minimum mapping problem was shown to be NP-complete for $K \geq 5$, but solvable in polynomial time for arbitrary K for tree networks in [4]. The general case is still open for $K = 2, 3$, and 4. A variation of the mapping problem is the duplication free mapping problem, where every node in the network is covered by exactly one LUT in the mapping solution. Duplication free area or delay minimum mappings can be obtained in polynomial time [1].

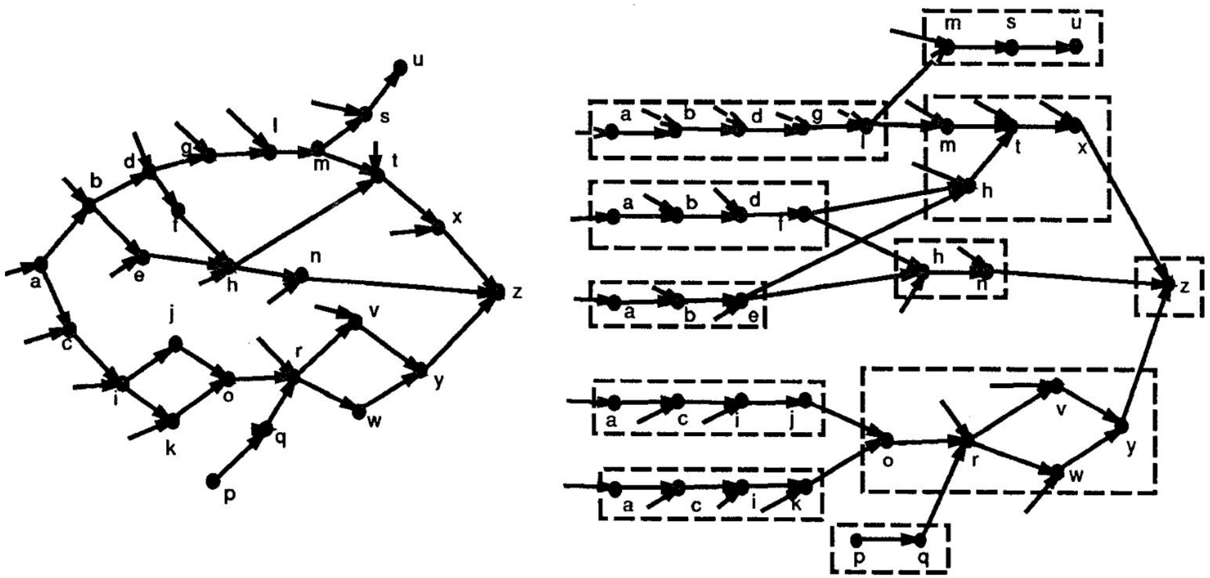
The main result of this paper is a polynomial time algorithm that computes the simultaneous area and delay minimum mapping solutions for K-input LUTs, if all nodes in the input network have *exactly* K inputs. We call such networks K-exact. This result contrasts with the NP-completeness of the area minimum mapping problem

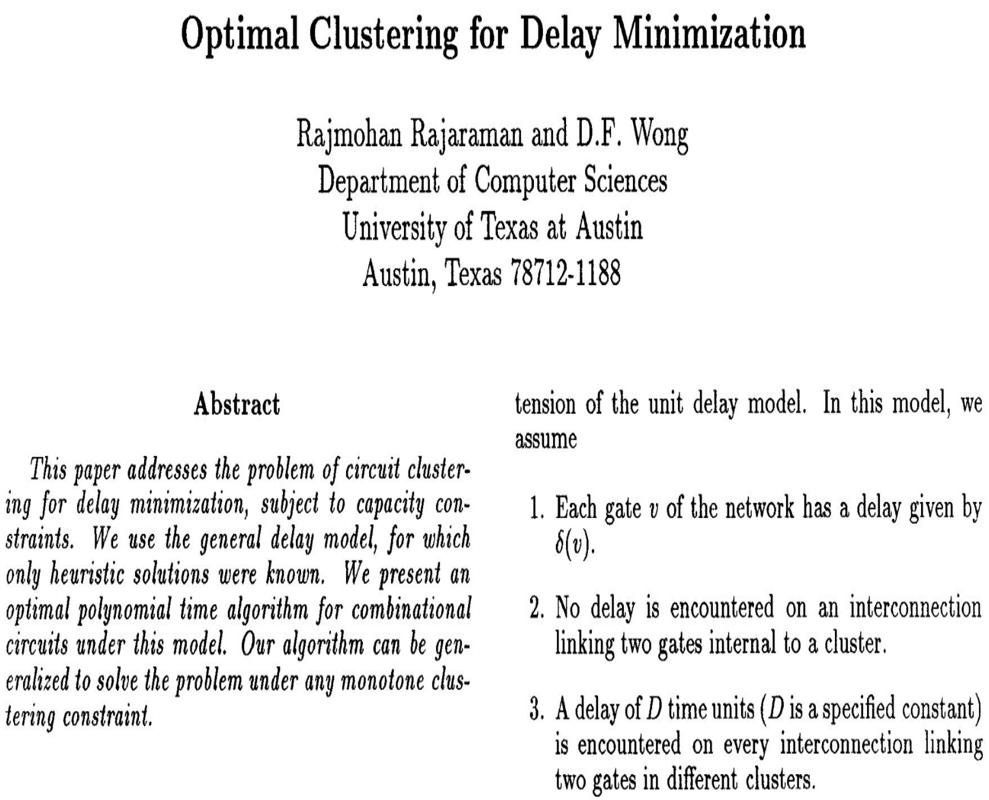




2-Exact Network

Optimal Clustering for Delay Minimization





Optimal Clustering for Delay Minimization

Rajmohan Rajaraman and D.F. Wong
Department of Computer Sciences
University of Texas at Austin
Austin, Texas 78712-1188

Abstract

This paper addresses the problem of circuit clustering for delay minimization, subject to capacity constraints. We use the general delay model, for which only heuristic solutions were known. We present an optimal polynomial time algorithm for combinational circuits under this model. Our algorithm can be generalized to solve the problem under any monotone clustering constraint.

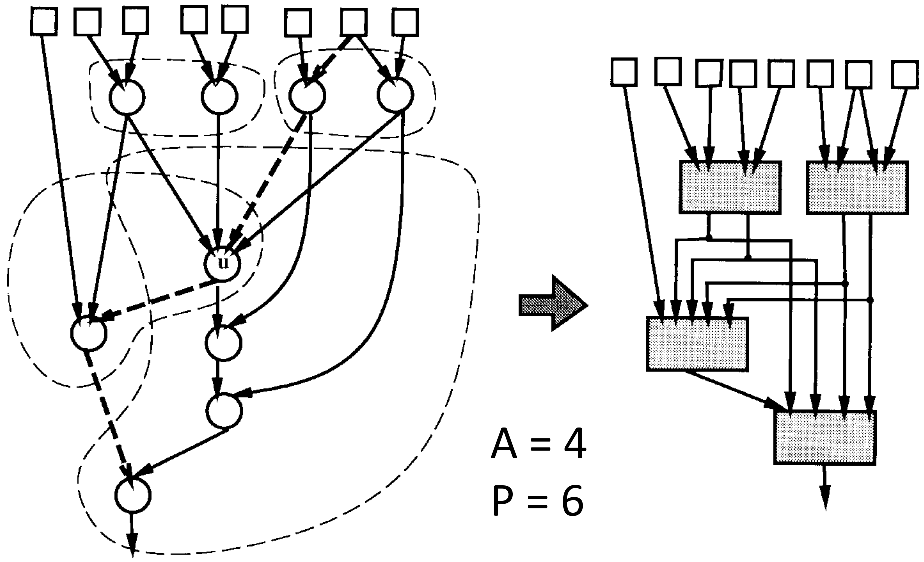
tension of the unit delay model. In this model, we assume

1. Each gate v of the network has a delay given by $\delta(v)$.
2. No delay is encountered on an interconnection linking two gates internal to a cluster.
3. A delay of D time units (D is a specified constant) is encountered on every interconnection linking two gates in different clusters.

DAC 1993

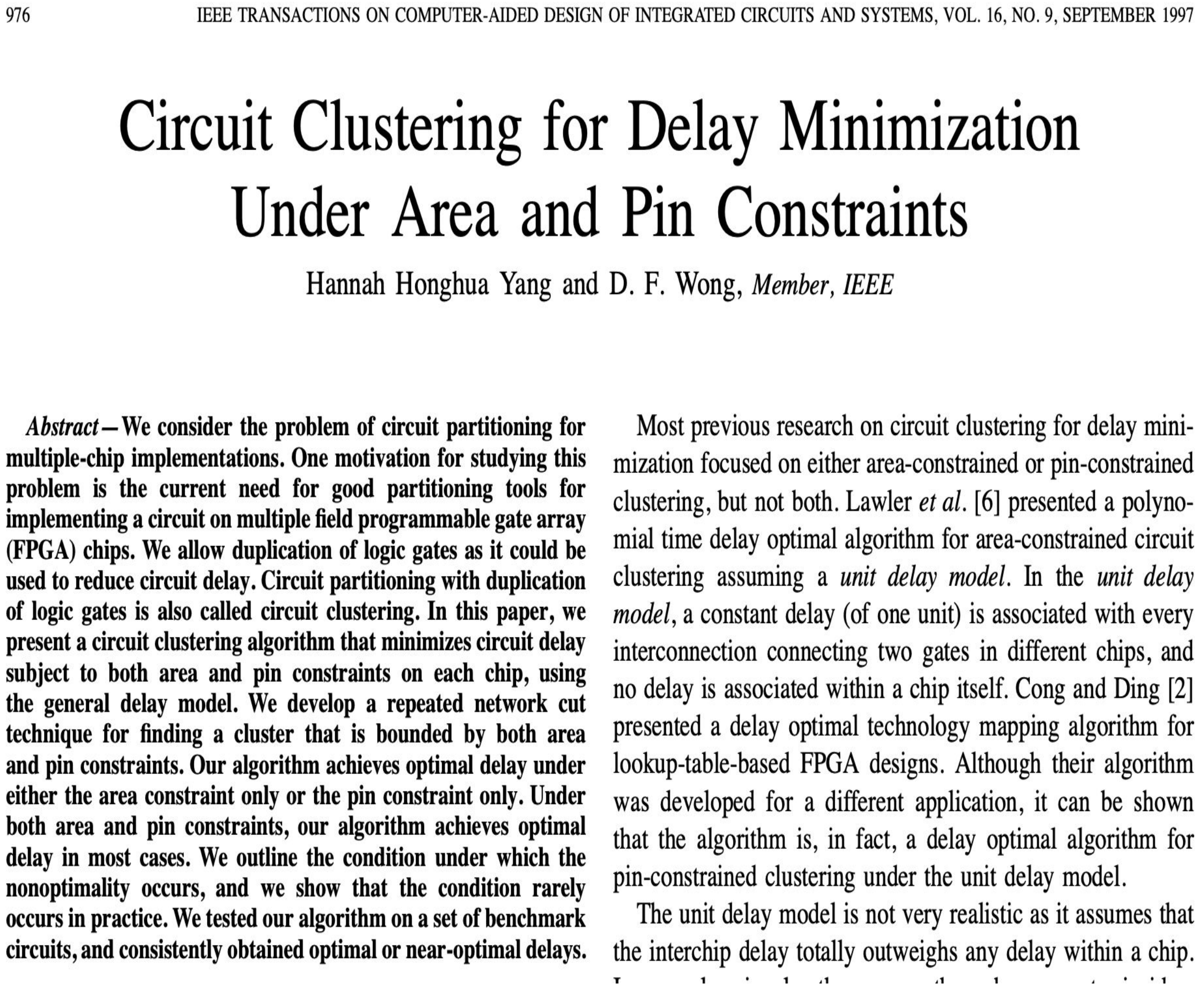
- Gate delay of d(v) for each gate v
- No delay for interconnections within a cluster
- A constant delay D for inter-cluster connection
- A constant area constraint for each cluster
- Gate replications allowed
- Polynomial time optimal algorithm for delay minimization

Clustering with Area and Pin Constraints



A = 4
P = 6

- Extension of Rajaraman & Wong (DAC 1993)
- Gate delay d(v), inter-cluster delay D, etc.
- Area and Pin constraints
- Optimal for either pin constraint or area constraint
- Perform well experimentally for both pin and area constraints


976 IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 16, NO. 9, SEPTEMBER 1997

Circuit Clustering for Delay Minimization Under Area and Pin Constraints

Hannah Honghua Yang and D. F. Wong, *Member, IEEE*

***Abstract*—We consider the problem of circuit partitioning for multiple-chip implementations. One motivation for studying this problem is the current need for good partitioning tools for implementing a circuit on multiple field programmable gate array (FPGA) chips. We allow duplication of logic gates as it could be used to reduce circuit delay. Circuit partitioning with duplication of logic gates is also called circuit clustering. In this paper, we present a circuit clustering algorithm that minimizes circuit delay subject to both area and pin constraints on each chip, using the general delay model. We develop a repeated network cut technique for finding a cluster that is bounded by both area and pin constraints. Our algorithm achieves optimal delay under either the area constraint only or the pin constraint only. Under both area and pin constraints, our algorithm achieves optimal delay in most cases. We outline the condition under which the nonoptimality occurs, and we show that the condition rarely occurs in practice. We tested our algorithm on a set of benchmark circuits, and consistently obtained optimal or near-optimal delays.**

Most previous research on circuit clustering for delay minimization focused on either area-constrained or pin-constrained clustering, but not both. Lawler *et al.* [6] presented a polynomial time delay optimal algorithm for area-constrained circuit clustering assuming a *unit delay model*. In the *unit delay model*, a constant delay (of one unit) is associated with every interconnection connecting two gates in different chips, and no delay is associated within a chip itself. Cong and Ding [2] presented a delay optimal technology mapping algorithm for lookup-table-based FPGA designs. Although their algorithm was developed for a different application, it can be shown that the algorithm is, in fact, a delay optimal algorithm for pin-constrained clustering under the unit delay model.

The unit delay model is not very realistic as it assumes that the interchip delay totally outweighs any delay within a chip.

TCAD Sept 1997; ED&TC 1995

The End