

An Overview of Placement and Routing Algorithms for Multi-Chip Modules

Subhomoy Chattopadhyay, Donald W. Bouldin and Peyman H. Dehkordi

Electrical & Computer Engineering

University of Tennessee

Knoxville, TN 37996-2100

Correspondence Author : Prof. Donald W. Bouldin

TELEPHONE : 615-974-5444

FAX : 615-974-5492

Email : bouldin@microsys1.engr.utk.edu

Abstract

Multi-chip modules are now required to achieve higher system speed and greater density than the traditional single chip packages mounted on printed circuit boards. Algorithms for placement of bare dies and routing of their interconnections on MCM substrates are reviewed in this paper. Comparisons are given to point out the strengths and weaknesses of each approach. This information can assist researchers in identifying those areas which need improvement and application designers in selecting the most appropriate algorithm for a specific application.

Keywords : Placement, routing, multichip, modules, algorithms, die

Contents

1	Introduction	4
2	Physical Design of MCMs	5
3	MCM Placement Algorithms	7
3.1	Performance-Driven Placement for MCMs	7
3.2	Conventional MCM Placement Approaches	8
3.3	Placement Using Multiple Design Criteria	9
3.4	Comparison of Different Placement Approaches	10
4	MCM Routing Algorithms	10
4.1	Introduction to the MCM routing problem	10
4.2	Maze Routing	11
4.3	Multiple-Stage Routing	12
4.4	Integrated Pin Redistribution and Routing	13
4.5	Gridless MCM Router based on Rubber-band Sketches	15
4.6	Performance-Driven Global Routing	16
4.7	Crosstalk Minimization Routing	21
4.8	Three-Dimensional Routing	22
4.9	Comparison of the Different Routing Algorithms	23
5	Conclusions and Scope for Future Work	23
6	Acknowledgments	26

List of Figures

1	<i>MCM Physical Design - Flow</i>	35
2	<i>Some Delay Models</i>	36
3	<i>A - Tree vs Steiner Tree</i>	37
4	<i>Zero - Skew Tree Construction</i>	38
5	<i>A tapered interconnect</i>	39
6	<i>A Multi Stage Clock Tree Distribution</i>	40

List of Tables

1	Comparison of Different MCM Placement Algorithms	41
2	Comparison of MCM Routing Algorithms, Part I	42
3	Comparison of MCM Routing Algorithms, Part II	43

1 Introduction

MCM physical design consists of two operations: placement and routing. *Placement* refers to the positioning of unpackaged or bare dies on the MCM substrate such that one or more figures of merit are optimized. Typical criteria include netlength minimization, proper heat distribution and overall system timing performance. *Routing* is the task of interconnecting the dies on the substrate such that one or more figures of merit are optimized. For routing, typical optimization criteria include netlength, via, layer or crosstalk minimization.

These two operations are mutually dependent, but the size and complexity of practical problems generally dictate that each one be solved separately. Each operation is order-dependent in that the placing of one die or the routing of an interconnection leave fewer options for the remaining dies to be placed or interconnections to be routed. Thus, those placements or interconnections which are considered to have a greater impact on the optimization criteria are performed first.

MCM physical design requires consideration of many issues which cannot be handled satisfactorily with computer-aided design tools developed for integrated circuits or printed circuit boards [49, 42, 12, 16]. Factors like the number of layers in the MCM substrate, the type of bonding and the type of substrate (ceramic, laminate or deposited) must be an integral part of the optimization procedure [39]. Considering the packaging issues early in the design cycle leads to better physical design and overall superior system performance [12].

An overview of the MCM physical design process is presented in the next section. In Section 3, MCM placement algorithms are detailed along with a comparison of their strengths and weaknesses. In Section 4, MCM routing approaches are presented and compared in a similar manner. The final section presents some general conclusions and outlines some areas for future

research.

2 Physical Design of MCMs

Figure 1 illustrates the operations involved in MCM physical design with respect to the overall design flow. The top-level hardware description language (VHDL or Verilog) is synthesized into a structural netlist of library components and their interconnections. For a very large design the netlist can then be partitioned into a set of different dies based on certain constraints such as area, thermal, delay and testability. This step is known as MCM partitioning or system partitioning. The goal of partitioning is to improve the functionality and routability of the design. After the layouts for the individual dies have been obtained, the dies have to be placed and then routed on the MCM substrate.

The actual physical design process does not include system partitioning and begins with the placement process. The system partitioning step is eliminated in cases where off-the-shelf bare dies are used as is the case with most MCM designs of today. The first main step in the physical design process is *chip placement*. MCM placement involves placement of dies on the MCM substrate such that the objectives of proper heat distribution, minimization of interchip delays, noise and tolerant load distance are taken into consideration. However when the fabrication issues are considered, additional constraints in the form of net separation and via constraint become important. This is because the fabrication of densely routed designs may result in low fabrication yield [16] or a non-manufacturable design.

The next step in the physical design process is *pin redistribution* [19] which uses a chip layer to improve the substrate routability of the design. The pin redistribution problem can be stated as: after the placement of the dies on the MCM substrate, redistribute the pins using chip layers

such that the total number of pin redistribution layers, crosstalk and the maximum signal delay are minimized. Another objective of pin redistribution is to maximize the number of nets that can be routed in a planar fashion. During pin redistribution each pin is to be assigned to the nearest point on a grid in the pin redistribution layers. Pins are redistributed uniformly with sufficient spacing so that the connections between the nets and the pins can be done without any design rule violations.

The final step in the physical design process is *routing* which usually consists of *global routing*, *layer assignment* and *detailed routing* [42]. The global routing step assigns nets or wires to different routing regions. Whereas the router in the detailed routing step actually finds a geometric path for each net and routes the terminals connected by the net. The primary goal of the global routing step is to reduce the length of the global routing trees since this reduces the area and delay. On the other hand, the layer assignment problem, which is in itself an NP complete problem, is a step in the routing process where each net is assigned to an x-y layer pair subject to the feasibility of routing the nets on a grid. The layer assignment step is important as it determines the number of layers in an MCM and hence the cost and the cooling mechanism required.

The MCM routing problem is different from the VLSI routing problem primarily due to the larger number of layers available in an MCM (e.g. 60 versus 4). Also the total number of nets and the density of interconnections in an MCM are much larger necessitating the development of a different set of CAD tools for the MCM design process. However at the present time most of the CAD tools for MCM physical design are either repackaged PCB or IC design tools. MCM routing has been considered as a three-dimensional routing process by [41] because the routing is not only done in the two-dimensional plane but also in the vertical direction in the multiple

substrate layers. The number of nets in the design, the substrate size, the number of chips mounted on the MCM substrate and the number of layers will continue to increase progressively necessitating that the MCM routing problem be efficient. The main objectives of MCM routing are reduction of the number of routing layers and the reduction or elimination of crosstalk which finally leads to an improvement in the overall system performance.

3 MCM Placement Algorithms

The MCM die placement problem can be defined as below :

Given a set of chips (dies) C and a set of chip sites S , find a mapping $\phi : C \rightarrow S$ subject to the timing (delay), thermal and area constraints in such a way so as to minimize the number of layers needed for routing.

Hence some of the important goals for the MCM placement tool are: minimize the total area of the substrate, ensure proper heat distribution, minimize the total length of the wire needed for routing and ensure routability of the design in minimum number of routing layers. MCM placement is a difficult combinatorial problem as a good placement involves a tradeoff between these mutually conflicting constraints. Some of the different placement algorithms and approaches are described next.

3.1 Performance-Driven Placement for MCMs

In MCMs some of the netlengths for connection between bare dies can be so long that they have a resistance which is comparable to the resistance of the driver. Such resistances cannot be neglected during placement when the delay due to the nets is estimated. Few approaches

have been developed to perform a delay or performance-oriented placement. Performance-driven placement is very important since the interconnect delays form a major part of the system cycle time. Several performance-driven algorithms based on delay models have been developed [44, 1]. The timing constraints fix the upper bounds on the netlengths and these upper bounds guide the placement process. The delay in multi-terminal nets depends on the net topology as well as the length of the net. The net topology is difficult to estimate at the placement stage of the design process making accurate delay estimation difficult. One way to get over the problem is to consider global routing and placement simultaneously as proposed in [43, 5]. A resistance-driven placement algorithm for placement in multichip modules has been developed by [42]. In this approach the net delay (cost) is modeled as the combination of the delay contributed by the wire that forms the net and the delay contributed by the sink capacitances. The second contribution is calculated based on all driver-to-sink capacitances.

Some researchers have considered the system partitioning and placement as one composite problem [42, 6]. Partitioning-based placement methods tend to spread the wiring across the layout surface and thus produce very routable placements. However it is difficult to model a truly integrated partitioning and placement approach that deals with all the issues involved in system partitioning and also routing. This approach applies for those cases in which the design of the dies has not been finalized yet MCM routing considerations are incorporated.

3.2 Conventional MCM Placement Approaches

The traditional placement methods can be categorized into two groups: *constructive and iterative*. Constructive placement methods take a partial placement as the input and produce a complete placement as the output. On the other hand, iterative placement methods begin

with one initial guess for the placement and then refine this placement taking into consideration certain constraints to obtain a better placement. A detailed description of these techniques is found in [51]. In the iterative placement approach probabilistic search optimization algorithms like the simulated annealing or genetic algorithms are used and the process is iterated until some stopping criteria is satisfied. In each iteration the components are moved around or rotated, and if the new configuration is better than the previous one, then the new configuration is selected. A cost function is invoked to determine the relative merit of each configuration. Some of these iterative methods are described in [51, 13, 17, 7] Esbensen and Mazumder [15] have combined the genetic algorithm and simulated annealing algorithm to speed up the optimization search and obtain better placements compared to either algorithm alone. This approach has been tested for macrocell placement and can also be extended to placement of individual dies on the MCM substrate.

3.3 Placement Using Multiple Design Criteria

Placement of multiple dies on an MCM substrate is a non-trivial task in which multiple criteria need to be considered simultaneously to obtain a true multi-objective optimization. Most researchers have considered only one criteria or objective in the MCM placement algorithms. In most cases it is netlength minimization which results in overall area and delay minimization. An ideal placement tool should result in the smallest layout while conforming to the electrical and thermal requirements. By doing a placement considering both the thermal and netlength criteria simultaneously, it can be seen that the final netlength after optimization is slightly higher. However the heat spread on the MCM substrate is more uniform compared to the case when only netlength minimization is done alone.

3.4 Comparison of Different Placement Approaches

MCM placement is much more complicated than the conventional IC placement problem in VLSI. Even though the number of dies in a MCM is generally well below 100, many interrelated factors determine the final layout quality. Table 1 compares different MCM placement approaches. The netlength minimization approach as followed by different researchers is not sufficient. Accurate delay estimation to account for the resistance of the nets is necessary.

4 MCM Routing Algorithms

4.1 Introduction to the MCM routing problem

The MCM routing problem can be defined as follows:

Given a set of placed chips (dies) and a netlist interconnecting different pins on the chips, route all the nets in such a way such as to use the minimum number of routing layers and satisfy some constraints.

Some of the objectives that need to be considered for optimization are netlength minimization, crosstalk minimization, via minimization and meeting the manufacturability constraints that guarantee that the yield is high and the designs are routable.

As MCM technology develops the number of chips on the MCM substrate and the number of layers will increase dramatically causing the need for very efficient routing algorithms. For example an MCM with 100 chips and 63 layers has been reported in [50]. The main performance constraints involved in MCM routing are *delay, noise and manufacturability* constraints. Delay constraints are converted to netlength constraints in most cases and require consideration to ensure the the MCM functions properly at the desired clock frequency. Noise constraints are

converted to path separation and parallelization constraints, and they require consideration to avoid unwanted logic switchings. In [41] the author describes in detail the fabrication constraints that should be considered for MCM routing.

As the complexity of the MCM routing increases more computation time will be required as the solution space of the combinatorial optimization problem becomes larger. A distributed computing environment may be necessary to obtain a good solution in a shorter computing time. A concurrent search of the solution space may be necessary with the help of different machines used to configure a PVM (Parallel Virtual Machine) [31]. Better time heuristics can be used to obtain a better quality solution in a PVM environment or the same quality of the solution can be obtained in a much shorter execution time. When a randomized search algorithm like the simulated annealing is used, most of the computation time is spent in calculating the cost function. The cost function calculation can be computed in a set of parallel machines which constitute the PVM. The PVM environment works in a *master - slave* mode. The optimization part is done in the *master* mode using the main host and the cost function calculation goes on in parallel using the *slave* mode. Near-linear speedups are possible using this approach especially for designs having thousands of nets [13].

A variety of routing approaches have been developed in recent years and are described next.

4.2 Maze Routing

The most commonly used MCM routing approach is *maze routing* [42, 13]. This algorithm, which was originally developed by Lee [32], is very simple to use conceptually but suffers from many inadequacies since the quality of the maze routing solution is very sensitive to the ordering of the nets. There is no effective algorithm for determining a good ordering of the nets in general

and a large number of vias are required for routing the nets towards the end even though there are a large number of signal layers. There is also the requirement for large memory to run the algorithm. It is impossible to consider performance constraints such as delay, noise and fabrication constraints in the maze routing algorithm. The high memory requirement for solving large problems as well as the large execution times have made the maze router highly unsuitable for most MCM applications. However, the maze routing algorithm is the backbone of many commercial routers that are available for MCM designs.

4.3 Multiple-Stage Routing

In multi-stage routing, the problem is decomposed into several smaller subproblems. The whole routing cycle is broken up into *pin redistribution*, *layer assignment* and *detailed routing*. These steps were explained in detail earlier in Section 2. Pins on the die layer (top layer) are first redistributed evenly with sufficient spacing in such a way that connections between nets and pins can be done without any design rule violation on signal distribution layers. Then layer assignment is performed so that each net in an x-y pair of layers are assigned subject to the feasibility of routing the nets on a global routing grid in each plane pair. The problem of layer assignment is NP-complete and is discussed in detail in [25, 42, 51, 26, 19]. After layer assignment, the next step is to route the nets using the signal distribution layers. This step, which is known as *detailed routing* depends on the layer assignment step. The approach may use only one single layer or a layer pair. Detailed routing which has been presented earlier in this paper is discussed further in [28].

4.4 Integrated Pin Redistribution and Routing

In this approach instead of distributing the pins before routing, the algorithm redistributes the pins with routing in each layer. After routing on one layer the terminals of the unrouted nets are propagated to the next layer. This concept was developed by Khoo and Cong [9, 30] as follows.

In [9] Khoo and Cong present an integrated routing approach called SLICE for MCM routing. Instead of the usual approach of distributing pins prior to routing, the SLICE router redistributes pins along with routing in each layer. This algorithm follows a layer-by-layer planar routing approach. The algorithm tries to connect as many nets as possible in each layer and the nets which cannot be completely routed in a particular layer are partially routed and then taken over to the next layer after scanning the routing region from left to right. After completing the planar routing in a layer the terminals of the unconnected nets are distributed so that they can be propagated to the next layer without causing local congestion.

In [30] Khoo and Cong have described a general area multi-layer router for MCMs called “V4R” since it uses no more than 4 vias for routing each of the two-terminal nets and no more than $4(k-1)$ vias for each k terminal net. The routing approach is somewhat similar to that of SLICE but V4R operates on x-y plane pairs instead of considering a layer at a time after the whole routing grid has been *sliced* into a number of layers. There is no topological routing step in V4R unlike other MCM routers. The physical routing is generated directly in V4R. The 2-D routing grid is divided into *vertical channels* by vertical columns. A vertical column is defined by a grid line that contains at least one net terminal. Similarly the *horizontal channels* are defined. Every net is routed using one of the two types of routing topologies: Type-1 topologies which have three vertical segments and two horizontal segments and Type-2 topologies which have three horizontal segments and two vertical segments. In each column of the grid, V4R

executes the following steps: *horizontal track assignment of the right terminals, horizontal track assignment of the left terminals, routing in the vertical channel and extension of the routing to the next column.* V4R is able to achieve significant improvements over a 3-D maze router. For example V4R uses 44 % fewer vias and 2 % less wirelength compared to a maze router [30]. V4R has a much faster execution time compared to the maze router and is also very suitable for the increasingly dense MCM designs of today.

A new approach to multi-layer routing for MCMs with pin redistribution is given in [20] and [19]. This router controls various design constraints including coupling between vias and signal lines and discontinuities such as vias and bends. The router redistributes pins or pre-wired subnets uniformly over the MCM substrate using pin redistribution layers. Pin redistribution is very important in MCM design since it not only provides a global distribution for the pins congested in the chip site over the chip layer so as to ease the future routing difficulty, but also reduces the capacitive coupling between vias induced by many layers (up to 63 layers) by separating the pins far apart. The ultimate goal of the problem is to minimize the number of layers required to redistribute the entire set. For signal distribution, a mixed version of *single-layer routing* and *x-y plane-pair routing* techniques was proposed to establish a tradeoff between *circuit performance* and *design objective* instead of emphasizing area minimization only. The strategy supports all of the via types such as stacked (blind) vias and staircase vias. One strategy is to apply single-layer routing iteratively until $\alpha\%$ of the nets are routed, then route the remaining $(100 - \alpha)\%$ nets by x-y plane-pair routing process. This provides the designer with a tradeoff (e.g., between the number of layers and total number of vias) and shows the versatility of the proposed techniques.

4.5 Gridless MCM Router based on Rubber-band Sketches

Grid-based routing schemes are not suitable for large dense MCM designs like the MCM-D [42]. SURF software [11] employs a rubber-band sketch routing approach without an underlying grid. The rubber-band routing approach for an MCM consists of two stages: the creation of a multi-layer rubber-band sketch followed by a conversion of the rubber-band sketch to a physical routing. It uses hierarchical top down partitioning approach to perform global routing of all nets together. The layer assignment is performed during the partitioning process to generate a routing that uses less number of vias and is not restricted to one layer one direction condition. The inputs to this router are a set of terminals, a set of obstacles and a set of wiring rules. The global routing approach followed by the authors of SURF uses two principles from the area of artificial intelligence mainly known as *the least commitment principle* and the notion of *maximal use of information*. The local routing is done one net at a time within the limits of the bin. Both the global router as well as the local router rely heavily on the data structure built on the constrained Delaunay triangulation. SURF generates a topological specification for flexible rubber-band routing. This approach is used for MCM routing across entire MCM substrates that have no channels. The SURF router builds on previous approaches in more than one way. Flexible bin interface specifications allow the global routing to be adjusted as more detailed information is available. By doing the layer assignment during the partitioning step and not restricting the results to the *one-layer-one-direction* constraint, fewer vias are used. This approach to routing represents a good balance between the need to make global decisions and satisfying the constraints imposed by the more detailed local levels.

4.6 Performance-Driven Global Routing

This approach has become one of the most important approaches to MCM routing recently since the role of interconnect in determining the performance of high-speed digital systems has been steadily increasing. It is estimated that interconnect delays account for 50% of the cycle time today, and this number will soon increase to 80%. This is especially true at the MCM level, since the wire-lengths are longer, and the off-chip drivers are bigger and contribute less to the overall delay. Thus, it is very important to consider interconnect delay in MCM routing.

In simplified terms, the delay of an interconnection net is a function of its topology, wire width and metal layer. Ideally, a performance-driven routing algorithm should be able to optimize all three of these simultaneously to achieve maximum performance. However, this is difficult to do in practice because of other constraints such as routability and problem complexity. Typically, the MCM routing problem is solved in three steps: **(1)** A global routing step determines the topology of each net on a coarse grid. Wire widths may be adjusted to minimize delays after determining the topology; **(2)** A layer assignment step assigns nets, or segments of nets, to different layers to optimize routability. Sometimes, layer assignment algorithms also consider crosstalk; **(3)** A detailed area router completes the routing on each layer, following the topology and width suggestions of the global routing.

This section considers the global routing step, and some aspects of the layer assignment problem. Before discussing the algorithms, it is helpful to gain an idea of the models used for the delay of an interconnection.

Delay Models

VLSI interconnect delay models either use the *lumped capacitor* delay model or the *RC tree*

model. The traditional “lumped capacitance” model for an interconnect, in which the delay of a net is proportional to its length, is not accurate enough at the MCM level. Since the effective resistance of off-chip drivers is comparable to the resistance of the interconnect wires, the wire resistance significantly affects the delay and cannot be ignored. Also the delay in a multi-terminal net is not only proportional to the netlength but also to the net topology. Hence we can say that there is a need for more accurate and elaborate delay models for MCMs compared to general VLSI design. The next few paragraphs outline this in greater detail. For a detailed description of the interconnect delay models for MCMs please refer to [42].

For reasonable accuracy, interconnects at the MCM level must be modeled as distributed RLC trees. In this model, each wire segment in the tree is represented by a distributed-RLC section, which consists of a distributed resistance and inductance in series, and a distributed capacitance in parallel to ground. For delay estimation purposes, each distributed section is typically approximated by splitting it into several lumped RLC sections. A first order model for the delay of an RLC tree is the Elmore model [14]. This model ignores the inductance component, and also ignores higher order effects such as “resistance shielding”. However, it has been demonstrated in [3] that this delay model has high “fidelity”, i.e., optimization done based on this model tends to lead to high-performance routes. Figure 2 shows the different delay models used. In [42] the author has developed a second order delay model for multi-terminal interconnects which demonstrates the effect of line inductance on second order delay models.

Higher-order models for interconnect delay can be computed efficiently using techniques such as AWE [37]. Programs such as RICE [38] are efficient enough to be used inside optimization loops.

Topology Optimization

When interconnect resistance becomes significant compared to the effective resistance of the driver, the interconnect delay of a multi-terminal net becomes a function of the topology of the routing. One of the first algorithms which attempted to optimize the topology for delay was presented in [34]. The algorithm iteratively constructs a tree for a multi-terminal net, by starting from a partially constructed tree (which is initially just the driver node), and connecting it to a sink node. The connection path is found using the “A” search algorithm, which is guided by a benefit value on each vertex in the routing graph based on estimates of the Elmore delay to all sinks in the tree, and the total wire length of the tree. An advantage of this algorithm is that it can be used on arbitrary routing graphs, so it can be used on channel graphs, or for area routing in the presence of obstacles. Many other performance-driven algorithms are restricted to the manhattan plane.

Another class of algorithms attempts to minimize Elmore delays directly during construction of the spanning tree for the net. The *Elmore Routing Tree (ERT)* and *Steiner Elmore Routing Tree (SERT)* algorithms [4] construct spanning and Steiner trees, respectively, using a greedy algorithm to minimize Elmore delay.

Using a slightly different delay model called the *dominant time-constant* model, a near-optimal algorithm for minimum-delay routing is presented in [10]. The algorithm is based on the concept of *Arborescence-trees*, which have the property that every driver-sink path is a minimum length path. An A-tree, in general, may have longer wire length than a minimum Steiner tree, but reduces delays when wire resistance is significant. It is shown in [10] that minimizing delay is equivalent to finding an A-tree with minimum wire length. Figure 3 shows the *A-trees* vs the Steiner tree.

An algorithm for finding near-optimal solutions is presented, and experimental results demonstrate delay reductions of up to 40% based on typical MCM parameters. The A-tree algorithm can also be extended to handle obstacles in the Manhattan plane.

Some performance-oriented algorithms do not directly minimize delay, but instead focus on objectives related to delay. For example, close examination of the Elmore delay model reveals that delay is related to the total wire length as well as the square of the source-sink path lengths. This observation is used in [8] to derive a *Bounded Radius Bounded Cost (BRBC)* algorithm, which takes a user-defined parameter ϵ and constructs a tree whose total wire length is at most $2(1 + 2/\epsilon)$ of the optimal Steiner length, and the longest path is at most $(1 + \epsilon)$ times the maximum source-sink distance. A *performance-oriented minimum rectilinear Steiner tree (POMRST)* algorithm [33] minimizes total wire length subject to constraints on the lengths of individual source-sink paths.

In addition to delay, it is sometimes important to minimize the *skew* of a tree, i.e., the maximum difference between the RC delays at different sink nodes in the tree. This is important in clock trees, since the clock signal should arrive at all clock nodes at the same time for correct operation of a synchronous digital circuit. Clock skew can be a significant part of the total cycle time. An algorithm for constructing a tree with *zero skew*, based on the Elmore delay model, was presented in [48]. The algorithm uses a recursive bottom-up strategy, building up a zero-skew tree (ZST) by recursively merging zero-skew subtrees. Figure 4 shows a zero skew tree construction. The original ZST algorithm has undergone improvements to minimize wire length [47] and to guarantee planarity of the resulting tree [27].

Wire sizing

When choosing the wire width of the route for a net, a careful tradeoff has to be made. If

minimum design rules are used for a long wire, the wire resistance will cause large delays. On the other hand, the wire cannot be made arbitrarily wide, since its capacitance will then dominate the delay. The best situation occurs when the wire width is allowed to vary in different parts of the net, being wider near the driver, and narrower near the sinks. This approach, known as *tapering* or *wire sizing*, can reduce RC delays significantly, while simultaneously achieving reduction in wire area (and hence power dissipation).

An optimal algorithm for wire sizing under the dominant time constant model is presented in [10]. A greedy algorithm is also presented in the same paper, which is much more efficient. A combination of the two algorithms produces an efficient optimal algorithm, which can reduce RC delays by as much as 50%.

For the Elmore delay model, a wire sizing algorithm is presented in [40]. A significant feature of this algorithm is that instead of simply minimizing the delay, it allows the user to specify delay constraints at the sink nodes. The algorithm then constructs a wire sizing solution using a sensitivity-based approach, such that the delay constraints are met and the wire area is minimized. This results in significantly better *engineering solutions*, since it is usually not important to make the delay smaller than the constraint, and attempting to do so causes the wire area to increase rapidly. By keeping the delay targets just 15% over the minimum delay, area savings of as much as 46% are observed.

Wire sizing can also be used to solve the skew minimization problem, using an approach similar to the ZST construction. Zero-skew algorithms based on wire length adjustments tend to generate solutions which are very sensitive to process variations: although the tree may have nominally zero skew, the actual skew due to process variations may be large. The algorithm of [35] uses wire sizing to achieve a *reliable* minimum skew solution, by varying wire widths.

4.7 Crosstalk Minimization Routing

Crosstalk refers to the parasitic coupling between neighboring wires due to the mutual inductance and capacitance effects. Crosstalk can increase delays or cause glitches which can cause incorrect operation. Several methods for controlling crosstalk have been proposed, including layer assignment, length control, variable spacing and wire sizing. The author in [13] has developed a crosstalk minimization router. It is claimed that the router developed in [13] not only minimizes the netlength used for routing but also places the wires that carry high frequency signals far apart. This reduces the electromagnetic interaction among the routed wires and thus reduces the *crosstalk*. The number of vias used in the router is controlled by a parameter known as the via control parameter. The input to the router is a *netlist* obtained after the placement of dies on an MCM substrate. Associated with each net is an integer value that represents the *switching activity* (frequency) of the signal carried by the net. Multi-terminal nets are decomposed into two-terminal nets and the router generates orthogonal routing. The routing is carried out by scanning the routing region horizontally or vertically from one end of the substrate to another. Each scan will route as many nets as possible in a pair (horizontal and vertical) of layers.

A layer assignment algorithm for crosstalk minimization is presented in [21], based on an efficient algorithm for maxcut k -color partitioning. Crosstalk minimization algorithms for channel and switchbox routing are presented in [29] and [45], respectively. These ideas could be adapted to MCM routing.

4.8 Three-Dimensional Routing

In [36] the authors have presented a new approach to MCM routing where they present the three-dimensional routing problem as smaller three-dimensional problems to achieve the best utilization of the three-dimensional routing space. The first phase of this approach involves routing distribution in the two-dimensional X-Y plane as well as in the Z direction. This approach ensures uniform routing over the MCM substrate and also satisfies the netlength constraints and manufacturing constraints. The algorithm is designed to take care of the following performance constraints: manufacturability, net-length, net-separation and via minimization. Manufacturability constraint should be such that the yield is maximum. When considering signal delays it is necessary to consider the delays in critical nets. The *netlength constraint* can be represented as,

$$l_i < \alpha_i \text{ for } n_i \in N$$

where, n_i represents each individual net, l_i represents the corresponding longest path from the source of n_i to the sink of n_i and α_i represents the length constraint

Net separation constraint primarily takes care of the crosstalk. The crosstalk between two lines can be minimized by ensuring that for each pair of nets the minimum separation is greater than a certain specified limit. The *via constraint* is introduced to minimize the maximum number of stacked vias for a net and satisfy the fabrication requirement. The via constraint is stated as,

$$v_i < \gamma \text{ for } n_i \in N$$

where the maximum number of stacked vias for a net is represented as γ and the number of stacked vias of net n_i at any given point is v_i . MCM routing is carried out in a three-dimensional routing space using a recursive formulation. The routing is completed in several phases which are

tiling, off tile routing, two-dimensional routing distribution, Z - dimension routing distribution, terminal assignment and tower routing. Tiling involves splitting the substrate area into smaller regions such that the denser regions are partitioned more compared to the less dense regions. The main function of the *two-dimensional routing distribution* is to see that the *manufacturability constraint* is satisfied. At the end of this phase the number of routing layers based on the routing congestion is estimated. The length of each net in the X-Y direction is also estimated. The objective of the *Z-dimension routing distribution* phase is to uniformly reduce congestion in the Z-direction. The terminal assignment phase is carried out by bipartitioning the substrate recursively.

The researchers who have developed this approach claim that the general objective of performance optimization using as few layers as possible has been satisfied using this three-dimensional approach.

4.9 Comparison of the Different Routing Algorithms

A lot of different routing algorithms for multi-layer MCM routing are available, each having some pros and cons. A comparative evaluation of the different algorithms is given in Table 2 and Table 3.

5 Conclusions and Scope for Future Work

A lot of work has already been done by different authors in the area of *placement* and *routing*. Most of the work in the area of MCM placement has focused on netlength minimization alone though some of the researchers have also considered the problem of proper heat distribution when placing the dies on the substrate. An automatic placement tool that considers both the

netlength minimization as well as the thermal criteria has also been developed. However there is a need to consider the wire resistance in delay estimation as the net delay is not just a function of the wirelength. More accurate algorithms need to be developed that would extend the concept of resistance-driven placement to RLC driven placement, to further reduce the interchip signal delays. Thermal constraints also have to be taken into account more carefully. Also it is no longer sufficient to do the substrate and IC placement in multi-chip modules individually in a standalone fashion [16]. Optimizing an MCM system requires close attention to the physical designs of the ICs and the substrate simultaneously. This type of approach will relax the routing constraints which are usually placed on the substrate when the IC designs have been completed before the substrate.

It is becoming clear to the IC and MCM design community that interconnects are no longer just “parasitics” - they are an integral part of the circuit design. Taking interconnects into account during circuit design can significantly improve system performance. To this end, new research, such as [18], is focusing on the problem of *simultaneous driver and wire sizing*. [18] presents efficient optimal algorithms for simultaneous driver and wire sizing for delay minimization, or delay and power minimization. Figure 5 shows a tapered interconnect.

Since clock distribution trees on an MCM can be very large, and heavily loaded, it is very difficult to drive them with a single driver. Typically, *buffers* are introduced at various nodes in the tree, to avoid having a single very large driver, and to keep the slope of the received waveform reasonably high. Such *multi-stage clock trees* introduce some interesting new optimization problems, such as finding the optimal locations for the buffers, minimizing the number of buffers, finding optimal sizes for the buffers etc. An algorithm for minimizing the number of buffers in a given clock tree subject to a clock period constraint is presented in [46]. The basic

algorithm is extended to handle upper bound constraints on the skew, and allow area and delay tradeoffs. Figure 6 shows a multi-stage clock tree distribution.

The disadvantage of constructing a matching-based hierarchical zero-skew clock tree is that the wires dictated by the matching edges in higher level of hierarchy are relatively long and they may introduce both a severe crosstalk noise due to buffer congestion of the generated tree topology. The local congestion problem can be eliminated by distributing the buffers over the plane with minimum impact on wirelength. By reducing the congestion of buffers, crosstalk delay will also be significantly reduced. Thus, the authors in [24] investigated the problem of reducing congestion, wirelength and clock skew during the growth of the clock tree. To take the three performance constraints into account simultaneously, the buffer distribution was formulated as a minimum length degree distributed spanning tree problem. An efficient solution to the problem is proposed in [24]. The proposed algorithm can be applied to MCM clock net routing. H-tree is preferable for clock distribution in the transmission line mode, since reflections and crosstalk can be minimized [2]. Thus, a clock tree routing scheme for hierarchical packaging system (e.g., MCM) would be as follows. An H-tree is used for inter-chip clock routing generating a set of clock sub-sources, each of which is inside each chip. Whereas pins in each chip are interconnected from the sub-source only inside the routing subregion using the proposed clock tree construction scheme.

The past few years have generated a large body of important results on interconnect optimization. However, there are still many areas which are unexplored. For example, performance-driven tree construction and wire sizing algorithms do not handle situations where a net has more than one driver. This frequently occurs on global signals like data and control busses, where performance is critical. Other important areas requiring further work are more accurate

modeling of transmission line effects and driver non-linearities. Currently, most routing algorithms use distributed RC delay models which do not take into account the inductive behavior of MCM interconnects. Furthermore, the driver is usually modeled as a single linear resistor, which can be a very inaccurate model. Finally, the various performance-optimization approaches - topology optimization, wire sizing and layer assignment - need to be unified into a single algorithmic framework and made to handle additional physical constraints such as congestion and routability.

Recently Field Programmable Multichip Modules (FPMCM) have been developed by researchers in [23], [22]. MCM technology has led to a great improvement in the performance and yield of FPGAs by bringing about an integration of MCM and FPGA technology. This new technology is called FPMCM (Field Programmable Multichip Module). Most of these FPMCMs use area I/O flip chip type of technology instead of the regular wire bond peripheral pads. This necessitates the need for a whole new set of tools for area pad assignment, placement and routing of FPMCMs.

6 Acknowledgments

The authors would like to express their special thanks to Prof. Jun Dong Cho from Sung Kyun Kwan University, Korea for his very valuable suggestions while writing this paper. We appreciate his help and encouragement. The authors would also like to express their thanks to Dr. M. Sriram of INTEL Corporation, Santa Clara for his very helpful suggestions during the course of writing this paper. His work in the area of MCM physical design has helped in a big way to identify some of the research done in the area of MCM placement and routing.

References

- [1] K. Chaudhury A. Srinivasan and E. S. Kuh. “ RITUAL : A Performance Driven Placement Algorithm ”. In *IEEE Transactions on Circuits and Systems*, 39(11):825-840, November 1992.
- [2] H.B. Bakoglu. “ *Circuits, Interconnections and Packaging for VLSI*”. Addison - Wesley, 1990.
- [3] K. D. Boese, A. B. Kahng, B. A. McCoy, and G. Robins. “ Fidelity and Near-Optimality of Elmore-Based Routing Constructions ”. Technical Report CS-93-14, Dept. of CS, Univ. of Virginia, Charlottesville, 1993.
- [4] K. D. Boese, A. B. Kahng, and G. Robins. “ High-Performance Routing Trees with Identified Critical Sinks ”. In *Proceedings of Design and Automation Conference*, pages 182–187, June 1993.
- [5] M. Burstein. “ A Non Placement-Routing Approach to Automation of VLSI Layout Design ”. In *Proc. Intl. Symposium of Circuits*, pages 234–244, 1989.
- [6] C. Harvatis, S. Mehrotra, P. Franzon. “ Topological Description and Interconnect Delay Modeling of Performance-Driven Partitioning and Placement of MCM Design ”. Technical Report NCSU-ERL-94-17, Dept. of Electrical Engineering, North Carolina State University, 1994.
- [7] J. Cohoon and W. Paris. “ Genetic Placement ”. In *IEEE Transactions on Computer-Aided Design of Circuits and Systems vol. CAD-6, no. 6*, page 956=964, November 1987.

- [8] J. Cong, A. Kahng, G. Robins, M. Sarrafzadeh, and C. K. Wong. “ Provably Good Performance-Driven Global Routing ”. *TCAD*, 11(6):739–752, June 1992.
- [9] J. Cong and K.Y. Khoo. “ An fast multilayer general area router for MCM designs ”. In *IEEE Transactions on Circuits and systems*, pages 841–851, November 1992.
- [10] J. Cong, K.-S. Leung, and D. Zhou. “ Performance-Driven Interconnect Proceedings of Design Based on Distributed RC Delay Model ”. Technical Report CSD-920043, UCLA, October 1992.
- [11] W.W. Dai. “ Topological Routing in SURF: Generating a Rubberband Sketch ”. In *Proc. IEEE Design Automation Conference*, pages 39–48, 1991.
- [12] P. H. Dehkordi and D. W. Bouldin. “ Design for Packageability - Early Considerations of Packaging from a VLSI Designer’s Viewpoint ”. In *IEEE Computer*, pages 76–81, April 1993.
- [13] G. Devaraj. “ *Distributed Placement and Crosstalk Driven Router for Multichip Modules* ”. MS Thesis, University of Cincinnati, 1994.
- [14] W.C. Elmore. “ Transient Response of Damped Linear Response with Particular regard to Wideband Amplifiers ”. In *Journal of Applied Physics*, pages 55–63, 1948.
- [15] H. Esbenson and P. Mazumder. “ SAGA - A Unification of Genetic Algorithm with Simulated Annealing and its Application to Macrocell placement ”. In *Proceedings of 7th International Conference on VLSI Design, Calcutta*, pages 211–214, January 1994.
- [16] Cadence Design Systems Inc. “ A Vision for Multichip Module Design in the Nineties ”. Technical report, Cadence Design Systems Inc., Santa Clara, CA, 1993.

- [17] C. King Iosupovici, A and M.A. Breuer. “ A Module Interchange Placement Machine ”. In *Proceedings of Custom Integrated Circuit Conference*, May 1984.
- [18] J. Cong and C.K. Koh. “ Simultaneous Driver and Wire Sizing for Performance and Power Optimization ”. In *ICCAD*, pages 206–212, November 1994.
- [19] J. D. Cho and M. Sarrafzadeh. “ The Pin Redistribution Problem in Multichip Modules ”. In *Mathematical Programming, North Holland - Special Issue on Applications of Discrete Programming in Computer Science*, pages 297–330, 1994.
- [20] K. F. Liao J. D. Cho, S. Rajee and M. Sarrafzadeh. “ A New Multilayer Routing System for High Performance MCM's ”. In *IEEE Transactions on Circuits and Systems, Vol 41, No 4*, pages 253–255, April 1994.
- [21] J. D. Cho, S. Rajee and M. Sarrafzadeh and M. Sriram and S. M. Kang. “ Crosstalk-Minimum Layer Assignment ”. In *IEEE Custom Integrated Circuits Conference*, May 1993.
- [22] R. Ramirez J. Darnauer, P. Garray and W. Dai. “ A Field Programmable Multichip Module ”. In *IEEE Workshop on FPPA for Custom Computing Machines*, pages 64–67, 1994.
- [23] V. Maheshwari J. Darnauer, T. Isshiki and W. Dai. “ FPMCM - An Integration of FPGA and MCM technology ”. In *IEEE Multichip Module Conference, MCMC - 1995*, pages 50–55, 1995.
- [24] J.D. Cho and Majid Sarrafzadeh. “ A Buffer Distribution Algorithm for High-Performance Clock net Optimization ”. In *IEEE Transactions on VLSI Systems, Vol.3 No. 1*, March 1995.

- [25] G. Vijayan J.M. Ho, M. Sarrafzadeh and C. K. Wong. “ Layer Assignment for Multichip Modules ”. In *IEEE Transactions on Computer Aided Design*, 9(12), pages 1272–1277, 1990.
- [26] G. Vijayan J.M. Ho, M. Sarrafzadeh and C.K. Wong. “ Layer Assignment for Multichip Modules ”. In *IEEE Transactions on Computer Aided Design*, pages 1272–1277, December 1990.
- [27] A. B. Kahng and C. W. A. Tsao. “ Low-Cost Single Layer Clock Trees With Exact Zero Elmore Delay Skew ”. In *ICCAD*, pages 213–218, November 1994.
- [28] M. Sarrafzadeh K.F. Liao and C. K. Wong. “ Single Layer Global Routing ”. In *Proceedings of Fourth Annual IEEE International ASIC Conference and Exhibit*, pages 14–4.1 – 14–4.4, 1991.
- [29] D. Kirkpatrick and A. L. Sangiovanni-Vincentelli. “Techniques for Crosstalk Avoidance in the Physical Proceedings of Design of High-Performance Digital Systems ”. In *Proceedings of ICCAD*, pages 616–621, November 1994.
- [30] K.Y. Khoo and Jason Cong. “ An Efficient Multilayer MCM Router Based on 4-via Routing ”. In *Proc. IEEE/ACM Design Automation Conference*, pages 590–595, June 1993.
- [31] Oak Ridge National Laboratory. “ PVM - 3 Users guide and Reference Manual ”. Technical report, Oak Ridge National Laboratory, Oak Ridge, Tennessee, 1991.
- [32] C.Y. Lee. “ An Algorithm for Path Connection and its Applications ”. In *IRE Transactions on Electronic Computers*, 1961.

- [33] A. Lim, S.W. Cheng, and C.T. Wu. “ Performance Oriented Rectilinear Steiner Trees ”. In *Proceedings of Design and Automation Conference*, pages 171–176, June 1993.
- [34] S. Prasitjutrakul and W. Kubitz. “ A Timing-Driven Global Router for Custom Chip Design ”. In *ICCAD*, pages 48–51, 1990.
- [35] S. Pullela, N. Menezes, and L. T. Pillage. “ Reliable Non-Zero Skew Clock Trees Using Wire Width Optimization ”. In *Proceedings of Design and Automation Conference*, pages 165–170, June 1993.
- [36] Q. Yu, J.L. Lai, S. Badida and N. Sherwani. “ A Performance Driven Three Dimensional Approach to MCM routing ”. Technical report, Dept. of CS, Western Michigan University, September 1993.
- [37] V. Raghavan, R. A. Rohrer, L. T. Pillage, J. Y. Lee, J. E. Bracken, and M. M. Alaybeyi. “ AWE-Inspired ”. In *Custom Integrated Circuits Conference*, pages 18.1.1–18.1.8, May 1993.
- [38] C. L. Ratzlaff, N. Gopal, and L. T. Pillage. “ RICE: Rapid Interconnect Circuit Evaluator ”. In *Proceedings of Design and Automation Conference*, pages 555–560, June 1991.
- [39] P. A. Sandborn and H. Moreno. “ *Conceptual Design of Multichip Modules and Systems* ”. Kluwer Academic Publishers, 1993.
- [40] Sachin S. Sapatnekar. “ RC Interconnect Optimization Under the Elmore Delay Model ”. In *Proceedings of Design Automation Conference*, pages 387–391, June 1994.
- [41] N. Sherwani. “ *Algorithms for VLSI Physical Design Automation* ”. Kluwer Academic Publishers, 1993.

- [42] M. Sriram and S.M. Kang. “ *Physical Design of Multichip Modules* ”. Kluwer Academic Publishers, 1993.
- [43] P.R. Suaris and G. Kedem. “ A Quadrisection Based Combined Place and Route scheme for Standard Cells ”. In *IEEE Transactions on Circuits and Systems*, 8(3), pages 234–244, 1989.
- [44] P.M. Vaidya T. Gao and C.L. Liu. “ A New Performance driven Placement ”. In *Dig. Tech. Papers, Intl. Conf on Computer Aided Design*, pages 44–47, 1991.
- [45] T. Gao and C. L. Liu. “Minimum Crosstalk Switchbox Routing”. In *Proceedings of ICCAD*, pages 610–615, November 1994.
- [46] G. E. Tellez and M. Sarrafzadeh. “ Clock Period Constrained Minimal Buffer Insertion in Clock Trees ”. In *Proceedings of ICCAD*, pages 219–225, November 1994.
- [47] Y.C. Hsu T.H. Chao and J.M. Ho. “ Zero Skew Clock Net Routing ”. In *Proceedings of Design and Automation Conference*, pages 518–523, June 1992.
- [48] R.S. Tsay. “ Exact Zero Skew ”. In *ICCAD*, pages 336–339, November 1991.
- [49] Rao R. Tummala. “ Multichip packaging - A Tutorial ”. In *Proceedings of IEEE*, pages 1924–1942, 1992.
- [50] R.R. Tummala. “ High Performance Glass-ceramic/copper Multilayer Substrate With Thin Film Redistribution ”. In *IBM Journal of Research and Development*, pages 889–902, 1992.
- [51] R. Vemuri. “ *Genetic Algorithms for Partitioning Placement and Layer Assignment for Multichip Modules* ”. Ph.D Dissertation, University of Cincinnati, 1994.

Biographies

Subhomoy Chattopadhyay Subhomoy was born in Calcutta, India. He obtained his BSEE and MSEE degrees in Electrical Engineering from the University of Calcutta and the Indian Institute of Technology, Madras in India in 1989 and 1991 respectively. He obtained his MS degree in Electrical and Computer engineering with specialization in VLSI and Multichip module design from University of Tennessee, Knoxville, TN, USA in 1995. He was also working part time on an US ARPA funded project on Multichip modules at University of Tennessee. He is presently employed at INTEL Corporation, Folsom, California. His area of interests include VLSI and Multichip module physical design and automation.

Donald W. Bouldin Dr. Don Bouldin is an IBM Professor of Electrical and Computer Engineering at the University of Tennessee, Knoxville. He received the B.E. (Electrical Engineering) from Vanderbilt University in 1967, the M.S. (Electrical Engineering) from Georgia Institute of Technology in 1968, and the Ph.D from Vanderbilt University, Nashville in 1975. Dr. Bouldin has authored more than 120 publications and been the Principal Investigator for nearly four million dollars of sponsored research, primarily with the nearby Oak Ridge National Laboratory, NSF and ARPA. Prof. Bouldin served two years as Chairman of the Technical Committee on VLSI for the IEEE Computer Society. He was instrumental in gaining IEEE approval for the IEEE Transactions on VLSI Systems and currently serves as its Editor-in-Chief.

Peyman H. Dehkordi Dr. Peyman Dehkordi is a Research Assistant Professor in Electrical and Computer Engineering at the University of Tennessee in Knoxville. He received his B.S., M.S., and Ph.D. in Electrical Engineering from University of Tennessee in Knoxville in

1981, 1985, and 1992. During 1982-1993, he developed real-time image-processing inspection systems for national and international markets. He has been the technical lead for over \$3 million in research and development for ARPA, SBIR, and private industries since 1992. He holds two US and one international patents. He is currently responsible for the technical leadership for the "Design for Packageability" ARPA project at University of Tennessee exploring design optimization techniques for multichip module designs.

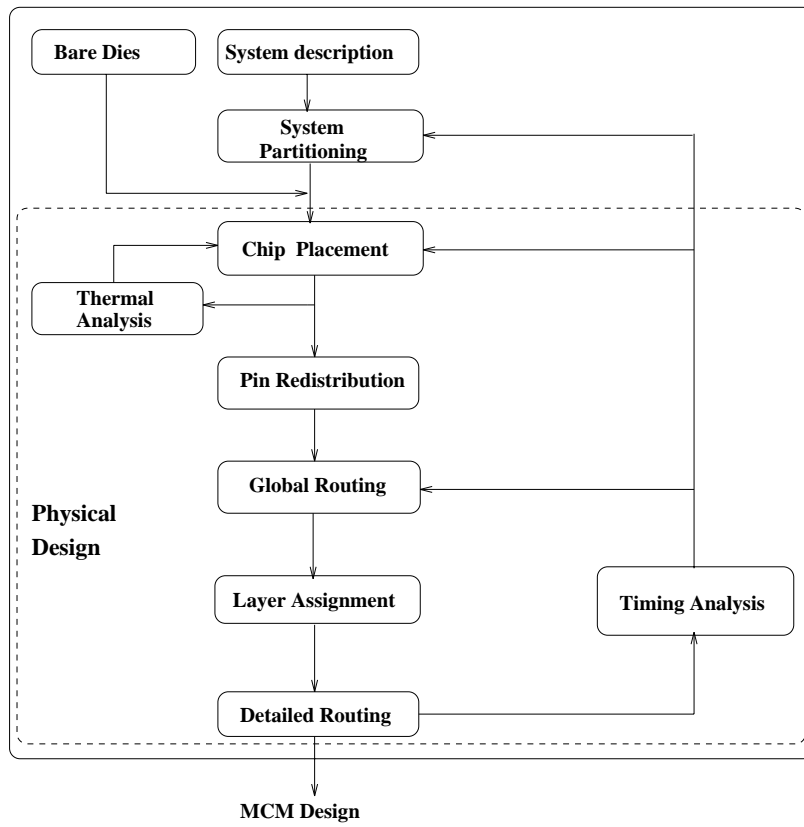


Figure 1: *MCM Physical Design - Flow*

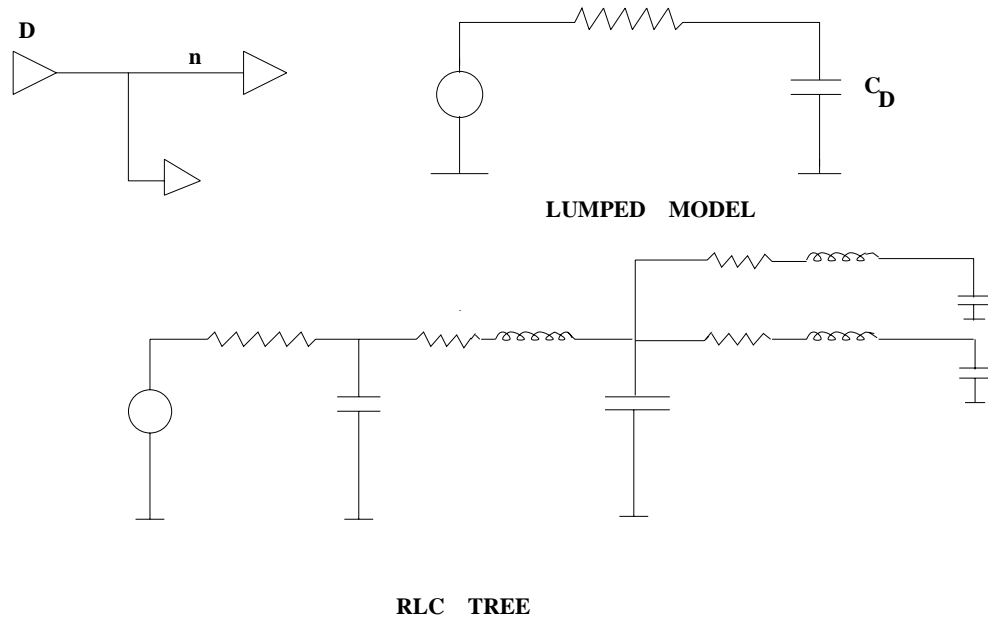


Figure 2: *Some Delay Models*

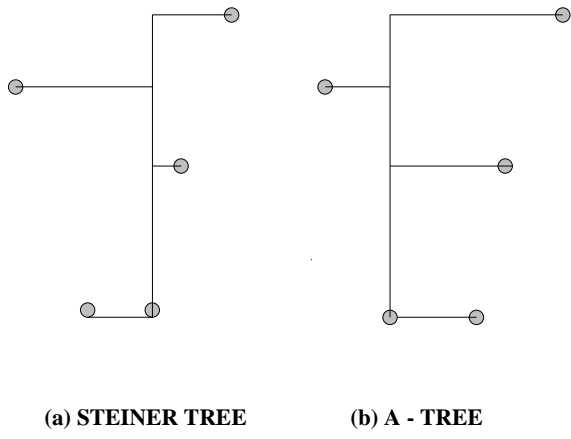


Figure 3: *A - Tree vs Steiner Tree*

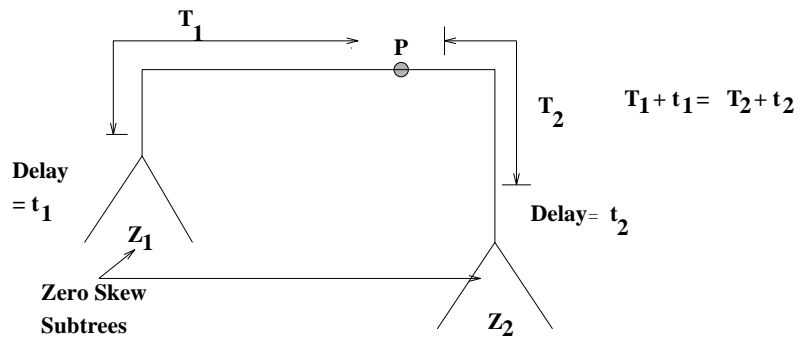


Figure 4: *Zero - Skew Tree Construction*

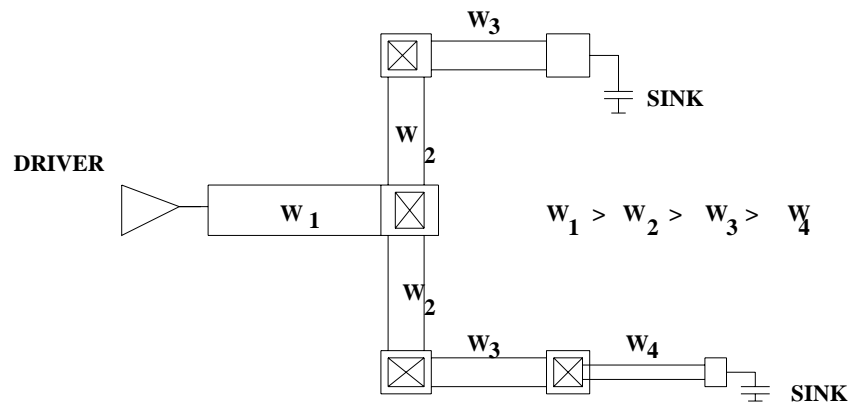
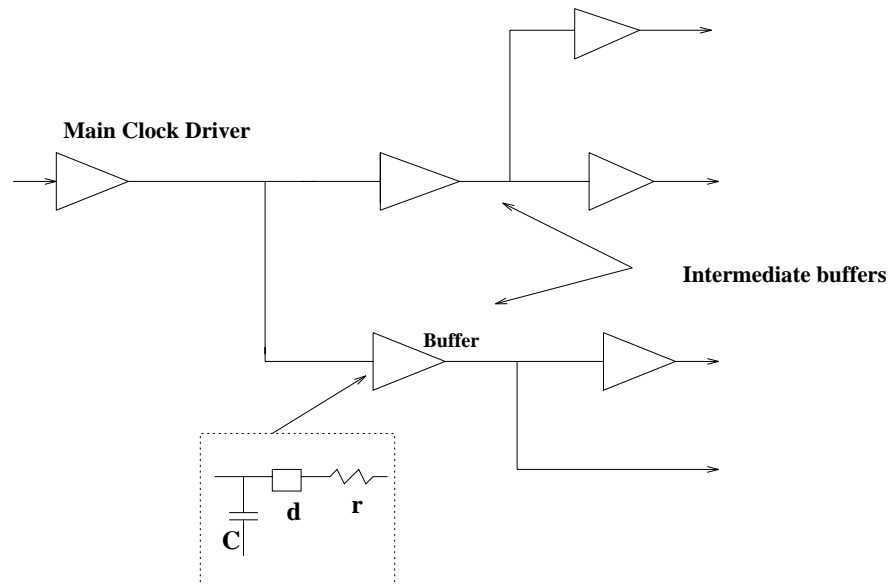


Figure 5: *A tapered interconnect*



C : Input Capacitance
d : Intrinsic delay
r : Output impedance

Figure 6: *A Multi Stage Clock Tree Distribution*

Author, Ref, Date	Netlength Minimization	Thermal Constraint	Performance Driven	Remarks
Franzon [6], 1995	Yes	No	Timing-driven	High probability of placements satisfying timing criterion
Vemuri [51],1994	Yes	Yes	No	Genetic algorithm iterative, gives a good solution
Devaraj [13],1994	Yes	No	No	SA-based, no thermal criteria, parallel implementation using PVM
Sriram [42],1993	Yes	No	Resistance-driven	Interchip signal delays reduced

Table 1: Comparison of Different MCM Placement Algorithms

Approach	Reference and Date	Netlength Minimization	Via Minimization	Layer Minimization	Crosstalk Minimization
Maze Router	[32], 1961	Yes	No	No	-
SLICE Router	[9], 1991	Yes	Yes	Yes	-
V4R Router	[30], 1993	Yes	Yes	Yes	-
Rubber-band Router	[11], 1991	Yes	Yes	Yes	Yes
Performance Driven	[42], 1993	Yes	Yes	Yes	Yes
Crosstalk Minimization	[13], 1994	Yes	Yes	Yes	Yes
3D routing	[36], 1993	Yes	Yes	Yes	No

Table 2: Comparison of MCM Routing Algorithms, Part I

Approach	Wiresizing	Runtime	Memory Requirement	Remarks
Maze Router	-	Large	Large	Heavy duty, inefficient
				large memory required
SLICE Router	-	Not large	Not large	Good for moderate sized designs
V4R Router	-	Not large	-	Improved version of SLICE router
Rubber-band Router	Yes	NI	NI	Gridless router, good for high density MCM designs
Performance Driven	Yes	Large	NI	Good and most accurate modeling of delays
Crosstalk Minimization	Yes	Fast	Not Large	Variable via and number of layers
3D routing	-	Not Large driven	Small	Efficient utilization of 3D routing resources

Table 3: **Comparison of MCM Routing Algorithms, Part II; NI = no information available**