# Fuzzy-clustering-based algorithm for circuit partitioning in standard cell placement

Jin-Tai Yan

*Indexing terms: Circuit layout cad, Fuzzy set theory*

The technique of circuit partitioning has been applied to standard cell placement for many years. A fuzzy-clustering-based algorithm is proposed to obtain a better two-way area-constrained partitioning for a partitioning-oriented standard cell placement. The proposed algorithm has tested several industrial circuit benchmarks, and the experimental results have shown that the algorithm obtains a better partitioning than the traditional F-M algorithm.

*Introduction:* In standard cell layout, the placement phase plays an important role in the automation of physical design. However, it is well known that the placement problem has been proved to be NP-hard [1]. Thus, the technique of circuit partitioning has been extensively proposed to develop a partitioning-oriented placement. In general, owing to the constraint of time complexity, two-way min-cut partitioning is always considered. In particular, the Fiduccia-Mattheyses (F-M) algorithm [2] is most often applied in the design of a standard cell placement [3, 4]. In Fig. 1, two-way circuit partitioning is applied to a partitioning-oriented standard cell placement.

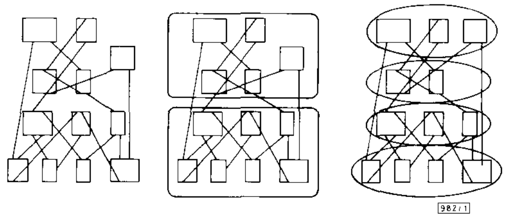


**Fig. 1** *Two-way circuit partitioning in a partitioning-oriented standard cell placement*

Fuzzy-clustering algorithms [5] have been extensively proposed for image processing and pattern recognition. Traditionally, based on the measure of geometrical distance, fuzzy-clustering algorithms will be applied to classify the image data on an image surface. In this Letter we extend these fuzzy clustering algorithms to a graph structure and further solve circuit partitioning for standard cell placement.

For circuit partitioning in a partitioning-oriented standard cell placement, first, a circuit netlist of standard cells will be transformed into an edge-weighted graph by a tree net model. Furthermore, based on the definition of the clustering distance on a graph structure, fuzzy graph clustering will be applied to obtain two groups of fuzzy memberships. Finally, according to these memberships and the information of cell areas, all the standard cells in a circuit netlist will be partitioned into two subcircuit netlists with a given area constraint.

*Problem formulation:* In standard cell layout, the height of each cell will be the same and the width of each cell may be different. Therefore, the area of each cell may be different. In general, based on the area consideration of standard cell layout, full area-balanced partitioning is more suitable for partitioning-oriented standard cell placement. However, owing to the area irregularity of standard cells and the assignment of feedthrough cells, area-constrained partitioning is more practical for partitioning-oriented standard cell placement. In two-way area-constrained partitioning, based on the net and area distributions of standard cells, a lower bound α and an upper bound β of any partitioning area will be estimated and given, where $0 \le \alpha \le 0.5$, $0.5 \le \beta \le 1$ and $\alpha + \beta = 1$. Hence, if a circuit netlist $C$ is partitioned into two subcircuit netlists $C_1$ and $C_2$, two-way area-constrained partitioning for a standard cell placement will be formulated as a partition $(C_1, C_2)$ of $C$ such that the cut of the partition $(C_1, C_2)$ is minimised with

$$\alpha \le \frac{\text{total\_area}(C_i)}{\text{total\_area}(C)} \le \beta \quad \text{for } 1 \le i \le 2$$

where total_area($C$) is the sum of areas of all the standard cells in $C$.

*Fuzzy-clustering-based partitioning algorithm:* In general, a circuit netlist is mapped as a hypergraph. Thus, circuit partitioning will correspond to hypergraph partitioning. It is well known that hypergraph partitioning is more difficult than graph partitioning. Hence, circuit partitioning is always solved by first transforming a hypergraph into a graph using a clique net model, that is, for any $p$-pin hyperedge $p > 1$, $p(p-1)/2$ complete connections of $p$ pins will be generated in the mapped graph. In general, based on the connection of $p$ pins, only $(p-1)$ edges in a tree net model are applied to maintain the connection of a $p$-pin hyperedge.

For a circuit netlist, we propose a tree net model to transform a multiple-pin net into a tree connection and assume that the cut contribution in a tree representation of a net must be obtained by an expected value of 1. Thus, for two-way partitioning of a $p$-pin net, the number of distributions of $p$ pins is $(2^p - 2)$. Assume that any distribution of $p$ pins is uniform; hence, the probability of any distribution of $p$ pins is $1/(2^p-2)$. If $i$ of $(p-1)$ connections are broken to separate $p$ pins into two different groups, the number of all the possible separations will be obtained as $2C_i^{p-1}$. Because the expected cut contribution of a $p$-pin net in a tree representation is 1, it is clear that

$$\frac{1}{2^p - 2} \sum_{i=1}^{p-1} 2C_i^{p-1} i w_p = 1$$

where $w_p$ is the edge weight of $(p-1)$ connections for a $p$-pin net in a tree net model.

Therefore, according to the previous equation, $w_p$ will be obtained as $w_p = 2/(p-1)$. By transforming all the hyperedges, a circuit netlist will be mapped by an edge-weighted graph. Hence, two-way circuit partitioning will be approximately obtained by two-way graph partitioning. In this Letter, two further phases are applied to obtain a two-way area-constrained partitioning after the graph transformation.

In phase 1, according to the mapped edge-weighted graph $G(V, E)$, a related clustering graph $G'(V', E')$ will be defined by modifying all the edge weights as $c'_{ij} = 1/c_{ij}$, where $V' = V$, $E' = E$ and $c_{ij}$ is the weight of the edge $\{i, j\}$. Furthermore, the clustering distance will be defined as follows: for any pair of vertices $i$ and $j$, the clustering distance $d'_{ij}$ between vertex $i$ and $j$ will be further obtained as

$$d'_{ij} = \begin{cases} c'_{ij} & \text{if } \{i, j\} \text{ is an edge in G} \\ \text{Short\_Path}(i, j) & \text{if } \{i, j\} \text{ is not an edge in G} \end{cases}$$

where Short_Path($s$, $t$) represents the sum of weights on the shortest path from vertex $s$ to $t$. Based on the definition of $d'_{ij}$ and fuzzy c-means clustering [5], as fuzzy graph clustering converges on all the fuzzy memberships, two groups of fuzzy memberships for all the vertices in G will be generated.

In phase 2, first, according to one group of fuzzy memberships, all the vertices $\{x_1, x_2, \ldots, x_n\}$ in G will be sorted decreasingly into a vertex list, $x^*_1, x^*_2, \ldots, x^*_n$. Furthermore, based on the areas of all the standard cells and the values of a lower bound α, and an upper bound β, two pairs of feasible area-constrained indices $(l_1, u_1)$ and $(l_2, u_2)$ for two clusters will be obtained by sequentially searching the vertex list such that

$$\sum_{i=1}^{l_1-1} A(x_i^*) \le \alpha(\text{total\_area}(C)) \le \sum_{i=1}^{l_1} A(x_i^*)$$

$$\text{and} \quad \sum_{i=1}^{u_1} A(x_i^*) \le \beta(\text{total\_area}(C)) \le \sum_{i=1}^{u_1+1} A(x_i^*)$$

and

$$\sum_{i=u_2+1}^{n} A(x_i^*) \le \alpha(\text{total\_area}(C)) \le \sum_{i=u_2}^{n} A(x_i^*)$$

$$\text{and} \quad \sum_{i=l_2}^{n} A(x_i^*) \le \beta(\text{total\_area}(C)) \le \sum_{i=l_2-1}^{n} A(x_i^*)$$

where $A(x)$ is the area of the standard cell representing vertex $x$. Finally, according to $(l_1, u_1)$ and $(l_2, u_2)$, a feasible partitioning index $(l,u) = (\max\{l_1, l_2\}, \min\{u_1, u_2\})$ will be generated for two-way area-constrained partitioning. Hence, a two-way min-cut partitioning will be further obtained by searching all the area-constrained partitioning.

As mentioned above, the fuzzy-clustering-based algorithm for circuit partitioning in a partitioning-oriented standard cell placement is as follows:

*Step 1*: Map a circuit netlist $C$ into a hypergraph $H(V, E_h)$.

*Step 2*: Transform $H(V, E_h)$ into an edge-weighted graph $G(V, E)$ by a tree net model.

*Step 3*:
(1) Initial an arbitrary two-way partitioning and establish a fuzzy matrix **U**.
(2) Compute all the clustering distance $d'_{ij}$ in G.

*Step 4*:
(1) Calculate the centres $v = (v_1, v_2)$ using **U** as follows:

$$v_i = x_j \quad \min_{x_j \in V} \left\{ \sum_{k=1}^{n} (u_{ik})^2 \text{Dist}(x_k, v_j)^2 \right\}$$
$$\text{for } 1 \le i \le 2 \quad x_j \in V$$

where $\text{Dist}(x_r, v_s)$ is the clustering distance between vertex $x_r$ and $v_s$.
(2) Calculate a new fuzzy matrix **U**′ using $v = (v_1, v_2)$ as follows:

If $(x_k \neq v_1 \text{ AND } x_k \neq v_2)$

$$u'_{ik} = \frac{d_{1k}^2 d_{2k}^2}{d_{ik}^2 (d_{1k}^2 + d_{2k}^2)} \quad 1 \le i \le 2,\ 1 \le k \le n$$

Else

$$u'_{ik} = \begin{cases} 1 & \text{if } x_k = v_i \\ 0 & \text{if } x_k \neq v_i,\ 1 \le i \le 2,\ 1 \le k \le n \end{cases}$$

(3) Compare **U** and **U**′; if $|u'_{ik} - u_{ik}| < \varepsilon$, for $1 \le i \le 2$, $1 \le k \le n$, then stop; otherwise, **U** = **U**′, and go to (1).

*Step 5*:
(1) Sort the vertex set $\{x_1, x_2, ..., x_n\}$ and construct an increasing vertex list $x^*_1, x^*_2, ..., x^*_n$ according to one group of fuzzy memberships $u_{1i}$, $1 \le i \le n$.
(2) Generate two pairs of feasible area indices, $(l_1, u_1)$ and $(l_2, u_2)$ and the feasible partitioning index $(l, u)$.
(3) Find a two-way min-cut partitioning by searching all the feasible area-constrained partitioning according to the partitioning index $(l, u)$ and compute the partitioning cut.

*Experimental results:* The proposed fuzzy-clustering-based algorithm has been implemented using standard *C* language and on a SUN workstation under the Berkeley 4.2 UNIX operating system. In this implementation, the $\varepsilon$ value in this algorithm is assigned as 0.01, and the proposed algorithm is run on each test benchmark 20 times with different initial partitioning.

For two-way circuit partitioning in a partitioning-oriented standard cell placement with an area constraint $(\alpha, \beta)$, the total cell area (TCA) is the sum of areas of the standard cells in a circuit netlist, and the area constraint $(\alpha, \beta)$ for any partitioning part $(PA_i)$ is as follows:

$$\text{TCA} * \alpha \le \text{PA}_i \le \text{TCA} * \beta$$
$$\text{for } 1 \le i \le 2,\ 0 \le \alpha \le 0.5,\ 0.5 \le \beta \le 1 \text{ and } \alpha + \beta = 1$$

**Table 1:** Comparison of experimental results with $(\alpha, \beta) = (0.4, 0.6)$

| Example | SA (10 runs) | | F&M (500 runs) | | PD | HGCEP | FCB (20 runs) | |
|---|---|---|---|---|---|---|---|---|
| | Best min-cut | Avg. min-cut | Best min-cut | Avg. min-cut | Min-cut | Min-cut | Best min-cut | Avg. min-cut |
| PrimGA1 | 36 | 38.9 | 37 | 86.5 | 39 | 38 | 36 | 38.4 |
| PrimSC1 | 41 | 63.8 | 39 | 88.6 | 39 | 38 | 38 | 41.3 |
| PrimGA2 | 131 | 168.5 | 146 | 397.0 | 123 | 119 | 115 | 124.7 |
| PrimSC2 | 128 | 156.4 | 157 | 402.3 | 120 | 119 | 115 | 126.7 |
| Test 02 | 90 | 109.0 | 107 | 126.7 | 92 | 83 | 79 | 85.4 |
| Test 03 | 59 | 86.4 | 81 | 145.8 | 58 | 58 | 53 | 59.6 |
| Test 04 | 53 | 69.5 | 44 | 46.0 | 43 | 44 | 44 | 45.5 |
| Test 05 | 53 | 107.8 | 42 | 44.9 | 42 | 47 | 42 | 43.8 |
| Test 06 | 74 | 85.5 | 74 | 180.7 | 62 | 47 | 45 | 52.7 |

For MCNC benchmarks in circuit partitioning, Table 1 shows the experimental results for two-way min-cut partitioning with $\alpha$ = 0.4 and $\beta$ = 0.6. In this Table, the simulated annealing (SA), Fiduccia-Mattheyses (F-M), primal-dual (PD), HGCEP and fuzzy-clustering-based (FCB) algorithms are evaluated. The results of SA, F-M, PD and HGCEP are from [6] and the results of FCB are from 20 runs.

 *17 October 1994*
*Electronics Letters Online No: 19950121*

Jin-Tai Yan (*Department of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan, Republic of China*)

**References**

1
2 FIDUCCIA, C.M., and MATTHEYSES, R.M.: 'A linear-time heuristic for improving network paritition s'. Proc. ACM/IEEE 19th Design Automation Conf., 1982, pp. 175–181
3 DUNLOP, A.E., and KERNIGHAN, B.W.: 'A procedure for placement of standard cell VLSI circuits', *IEEE Trans.*, 1985, **CAD-4**, pp. 92–98
4 CHO, H.G., and KYUNG, C.M.: 'A heuristic standard cell placement algorithm using constrained multistage graph model', *IEEE Trans.*, 1988, **CAD-7**, pp. 1205–1214
5 KIM, T., BEZDEK, J.C., and HATHAWAY, R.J.: 'Optimality tests for fixed points of the fuzzy c-means algorithm', *Pattern Recognit.*, 1988, **21**, pp. 651–663
6 SHIN, H., and KIM, C.: 'A simple yet effective technique for partitioning', *IEEE Trans. VLSI Syst.*, 1993, **1**, (3), pp. 380–386

## Mixed-mode Schmitt trigger equivalent circuit

S.R. Ramirez Chavez

*Indexing terms: Comparators, CMOS integrated circuits, Equivalent circuits, Mixed analogue-digital integrated circuits, Trigger circuits*

A circuit implementing the regenerative comparator or Schmitt trigger function is presented. While the traditional Schmitt trigger implements the hysteresis levels by means of positive feedback in an analogue loop, our circuit implements hysteresis by digitally processing the output of two comparators operating in open-loop mode. The threshold levels of the new circuit can be independently set and fine-tuned which is not the case in many traditional Schmitt trigger implementations.

*Introduction:* The regenerative comparator or Schmitt trigger was introduced by Otto Schmitt in the 1930s [1]. Since then, this circuit has been used to reduce the noise effects in triggering devices [2], analogue to digital conversion [3] and other applications.

Since its invention, the Schmitt trigger circuit has relied on changing the voltage or current threshold levels by means of positive feedback in the analogue loop. In their textbook, Millman and Halkias [4] discuss how this is done by means of a resistive voltage divider. Other voltage mode feedback circuits, which are more suitable for VLSI implementation, are discussed by Steyaert [5] and Dokik [6]. Schmitt trigger circuits with current feedback are discussed by Filanovsky [7] and Wang [8].

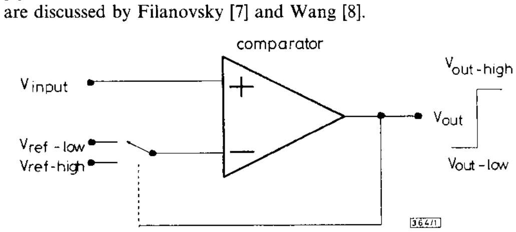


**Fig. 1** *Traditional voltage mode Schmitt trigger principle of operation*

*Principle of operation of traditional Schmitt trigger:* The basic principle for a voltage-mode feedback Schmitt trigger is shown in Fig.