Journal of Advanced Engineering Vol. 5, No. 4, pp. 347-355 / October 2010

基於線性規劃為基礎考量連線長度之 ECO 演算法
Integer Linear Programming-based ECO Approach with Wire Length Consideration

陳漢忠 [1] 黃信雄 [2] 謝財明 [*1]
H. C. Chen[1], H. H. Huang[2], T. M. Hsieh[*1]

摘要

本論文提出一以線性規劃為基礎的方法，將備用晶元選擇問題成功轉換成資源分配問題。本論文的最佳化目標是針對給定工程修改內容，搜尋緩衝器備用晶元以達成工程修改，並且最小化連線總長度。為了加速演算法的執行時間，本論文提出一以插入虛擬節點為考量之新搜尋方法，考量連線中點和所有備用晶元的座標，有效率搜尋最短連線總長度的晶元配置解。本論文所提的方法成功將電路特性轉換成線性方程組，連線組將能描述：每一工程修改問題僅能使用單一備用晶元、單一備用晶元僅能配置給單一之工程修改；若有些連線插入緩衝器依然時序違反，則再執行一次 ILP 線性模組。針對連線方程組使用商用軟體 Lingo 求得最小連線長度之晶元配置。實驗結果顯示：本論文所提線性規劃為基礎的方有效率且有效地減少工程修改所需之連線總長度。
關鍵詞：工程修改問題，線性規劃方程組，連線總長度，備用晶元選擇，虛擬點

Abstract

This paper identifies an integer linear programming-based approach with regards to the spare cells selection as the problem of resource allocation. The objective aims to search the spare cell assignment with minimization of total wire length. A novel wire length estimation method is proposed to accelerate the runtime of the proposed approach. In the proposed method, only the virtual node, i.e. the location of buffer insertion (middle point in this work) of the interconnection and the spare cells are taken into account for acquiring the spare cell assignment with the minimization of total engineering change order in length. If the timing violations still appear, the procedure will insert buffer again by performing the ILP formulations. The circuit characteristics, including the demand of each engineering change order and the supply of spare cell resource, could be successfully integrated into the ILP formulations. Experimental results showed that the proposed integer linear programming-based approach significantly minimized the total wire length.
Keywords: engineering change order, integer linear programming, total wire length, spare cell selection, virtual node

I. INTRODUCTION

Mask costs of the advanced technology increased exponentially. To reduce mask costs [1-3], many researches explored various methods for functional and timing changes. The available spare cells are commonly applied to perform engineering change order (abbreviated as ECO) to fix timing violation at later stages, such as placement or integrated circuits from the foundry [2, 4]. Most ECO approaches focus on the technology mapping and spare cells selection [5-6] to shorten time-to-market for chip bugs. Spare cells are not connected to components or subcircuits in a chip design, and are inserted to facilitate the chip debugging, timing and functional violations. To rewire the spare cells and terminals, a netlist is modified to form a new design. In this approach, the ECO which applied the spare cells to rewire the violated subcircuits truly shortened the time-to-marker.

Many ECO schemes explored to search for the available resources, such as buffers and gates, in order to fix timing violations or functional issues. Chang *et al.* [4]

[1] 中原大學資訊工程學系 [2] 龍華科技大學電子系
[*]Corresponding author. E-mail: hsieh@cycu.edu.tw
[1] Department of Information and Computer Engineering, Chung Yuan Christian University, Chung-Li, Taiwan, R.O.C.
[2] Department of Electronic Engineering, Lunghwa University of Science and Technology, Taoyuan, Taiwan, R.O.C.
Manuscript received 30 April 2010; revised 14 June 2010; accepted 23 July 2010

equally utilized the resource of spare cells by analyzing the spare cells in the chip. They obtained a much smaller delay cost by determining the density of spare cells and the regional demand of spare cells. Kuo *et al.* [7] integrated the spare cells selection with constant insertion technique to automatically determine the assignment of spare cells. Constant insertion technique, which tied some inputs to power and ground, produced the equivalent circuits. Chen *et al.* [8] presented the timing-driven algorithm to meet timing constraints by taking into consideration the gate sizing and buffer insertion. An efficient pruning technique was used to reduce the solution in search of spare cells. Most approaches applied the half-perimeter wirelength approach to consider the input/output terminals and spare cells. However, searching all spare cells of the bounding box is rather inefficient. Instead of searching the entire bounding box, it is efficient to search the small region formed by the location of buffer insertion and the spare cells. Moreover, some traditional methods sequentially assigned available spare cells and no papers have guaranteed that their approaches can obtain the spare cell assignment with the optimal length. We observed that the spare cell selections problem can be formulated and solved by using a linear programming-based approach to achieve the optimal spare cell assignment.

Many approaches provided the integer linear programming (ILP for short) based method to get the optimal solution in accordance with their goal. Generally, a problem is first formulated into the corresponding ILP formulations and then the ILP formulations are solved by using the commercial tools. Ekpanyapong *et al.* [9] improved the total power of circuit under time constraint by applying the integer linear programming-based approach. Huang and Cheng [10] proposed a low power-driven approach for optimizing the peak power consumption yielded from the ILP-based method. Davoodi and Srivastava [11] formulated the problem of low power-driven binding under resource constraints, which efficiently solved the problem resulted from the graph-based methods. Huang and Cheng [12] presented the ILP approach to optimize the power at the scheduling for high level synthesis. However, no existing ILP-based method has been adopted to solve the ECO problem.

To solve the disadvantages from previous works, the main contributions are described as follows. To the best of our knowledge, it is the first work that transfers ECO problem into the ILP-based formulation which effectively minimizes the ECO wire length. In addition, a novel wire approach, which searches the small region formed by the spare cells and buffer insertion locations instead of the whole bounding box region covering the terminals and spare cells, is applied to accelerate the runtime.

The organization in the paper is as follows. Section II discusses three types of the engineering change order problem. Section III describes our motivation to minimize the total wire length of ECO and defines the ECO problem in the paper. Section VI illustrates the ILP-based approach to minimize total wire length. Experimental results are shown in Section V and conclusions are provided in Section VI.

II. THREE TYPES OF ENGINEERING CHANGE ORDER

Spare cells are the unused redundant cells which do not connect with any components in the design. Due to many unused locations of the chip core after placement, the chip designer usually inserts the dummy cells, such as the spare cells and gate array [13] which is a new repaired cell structure. According to the design experiences, most designs only make use of 70~80% utilization during the placement. The percentage of the dummy cells, such as the spare cells and gate array, accounts for about 20%. Therefore, the ratio between the used cells and spare cells is approximately 4:1. Hence, how to manage the spare cells to the ECOs properly while minimizing additional wire length is quite difficult and significant.

Generally, ECO techniques which utilize spare cells to fix chip bugs can shorten the time-to-market after chip tape-out. ECO types are divided into three types: buffer insertion for wires, replacement for gates and rewiring for sub-circuit which are illustrated by Figure 1 to 3, in these figures symbols representations are shown by NAND gate, AND gate, OR gate, spare cell (buffer), interconnect, timing violation.

Figure 1 is used to illustrate the type 1 ECO. Figure 1(a), the 2-pin net with timing violation must insert buffer to fix the violation. Figure 1(b) is the circuit after buffer insertion. We apply Figure 2 to explain type 2 ECO. The NAND gate with circle symbol in Figure 2(b) needs to be replaced by the AND function. Figure 2(b) is the final circuit after gate replacement. Type 3 ECO is illustrated in Figure 3. A chip designer intends to change the original function $f_1 = \bar{x} + \bar{y} + zw$ in Figure 3(a). After rewiring the sub-circuit in Figure 3(a), the final circuit is shown in Figure 3(b).

In the paper, we only emphasize on type 1 ECO which inserts the buffers to the timing-violation interconnections by using single type buffer. The objective

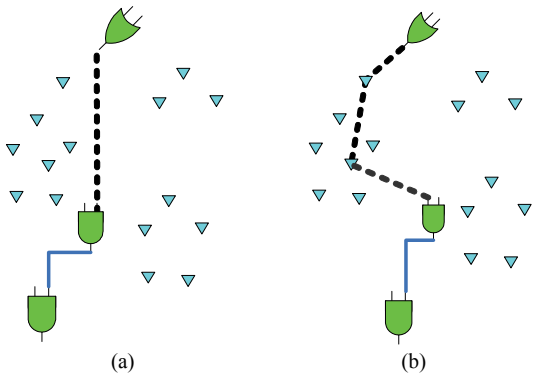


Fig. 1 Type 1 ECO (Timing change)

aims to optimize the total wire length for inserting buffers in order to fix timing violation. We observe that longer wires lead to large additional delay. Hence, the additional wire length due to spare cell selection is larger than the user-defined length and we do not perform ECO to rewire the netlist. Therefore, the huge penalty of additional wire length due to ECO issues can be avoided. Generally, there are k buffers insertion to fix timing violations for each 2-pin net. For example, as shown in Figure 1, the net with dotted line is timing violation in Figure 1(a) and we fix the timing violation by inserting two (k=2) buffers to the interconnection in Figure 1(b). When the timing violation is not serious, the number of buffer insertion is few. To simplify and evenly distribute the spare cells to the ECOs, the number of buffer insertion for each 2-pin net is set to be one (k=1) for this work. If the timing violation still appears after buffer insertion, the circuit will be formulated into another ILP formulation. Until all nets satisfy the timing specification, the procedure of iteratively ILP formulations will terminate.

III. PRELIMINARY

In this section, an insertion-based selector, motivation and problem definition are discussed in detail. The length calculation and buffer insertion are reviewed followed by the description of an insertion-based selector. All symbols used in the paper are shown in Table 1.

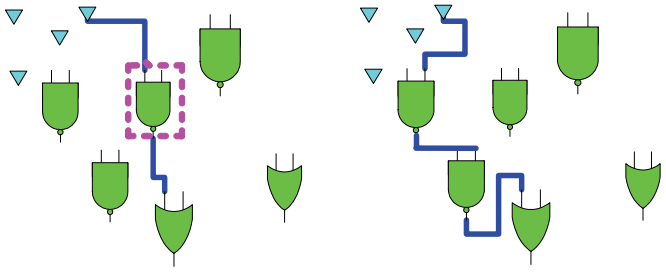


Fig. 2 Type 2 ECO (Functional change): (a) NAND gate, (b) AND gates

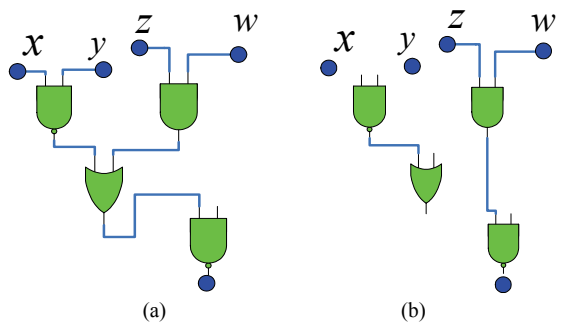


Fig. 3 Type 3 ECO (Rewiring): (a) $f_1 = \overline{x} + \overline{y} + zw$, (b) $f_2 = zw$

Table 1 Symbols used in the paper

SYMBOL	MEANING
r	Location of buffer insertion for a interconnection from input to output terminal
k	Number of buffer insertion for each 2-pin net
$\{t_1, t_2, \ldots, t_q\}$	A set of input and output terminals of the netlist
$\{s_1, s_2, \ldots, s_m\}$	A set of spare cells
$\{eco_1, eco_2, \ldots, eco_n\}$	A set of engineering change order for the given netlist
$\{v_1, v_2, \ldots, v_n\}$	A set of virtual nodes to select the spare cells
$span\_x(i,j)$	Half-preimeter wire length (over x-asix) of bounding box
$span\_y(i,j)$	Half-preimeter wire length (over y-asix) of bounding box
$d(i,j)$	Insertion-based distance from virtual node v_i to spare cell s_j
$b(i,j)$	Binary variable to represent that the virtual node v_i is assigned by the spare cell s_j
$len\_eco(i)$	Wire length for ECO eco_i
all_len_eco	Total wire length for all ECOs

The length for a multi-terminal net is based on the calculation of half perimeter wire length (HPWL for short) of the minimization bounding box. For net i, the wire length is estimated as the follow.

$$d(i,j) = span\_x(i,j) + span\_y(i,j) \quad (1)$$

where $span\_x(i,j)$ and $span\_y(i,j)$ denote the minimum length of a bounding box covering the input/output terminals of the net i and spare cell j, respectively. An example shown in Figure 4 is used to illustrate the calculation of HPWL, in which HPWL from (50, 80) to (150, 20) is 160 (=100+60).

The location of buffer insertion has been explored in many researches in fields such as the feasible region [14] and independent feasible region [15]. In the paper, we defined the location of buffer insertion as virtual node v_i. The relative research to estimate the location of buffer insertion was time consumption. The location of buffer insertion was defined as the r ratio of the wire length from the source to target. To simplify, the location of buffer insertion was set to the middle node (r=0.5) of the timing-violation interconnection. In the following subsection, the concept of the virtual node with the user-defined ratio r is discussed in detail.

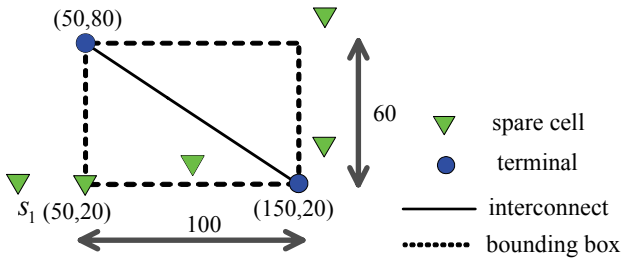


Fig. 4 Calculate the wire length by using HPWL

1. An Insertion-based Selector

Based on the foregoing discussion, the buffer insertion location for each ECO is defined as the *eco node* which denotes a virtual node of a 2-pin net. For each eco node, the spare cell with the shortest wire length is selected by using the HPWL. The insertion-based selector which searches a small region formed by virtual node and the spare cells, efficiently obtains the spare cell.

Figure 5 is used to illustrate the procedure of the insertion-based selector. The circle and square denote the terminals and spare cells, which include the assigned and unassigned cells, respectively. The triangle is the original connection in the circuits. For net ($k=1$) in Figure 5(a), the wire length between the virtual node (r=0.5) and spare cell S_3 is computed as follow,

$$d(1,3) = span\_x(1,3) + span\_y(1,3) = 50+30 = 80 \quad (2)$$

For net ($k=2$) in Figure 5(b), the "box1" is valid because the "box1" contains two spare cells. In contrast, the "box2" is invalid because the "box2" contains only one spare cell.

Summary, the wire length from the virtual nodes to spare cells are calculated by using the insertion-based selector. We integrate the related information into the ILP formulation discussed in the Section III.

2. Motivation

We use the example depicted in Figure 6 to illustrate our motivation. Due to the long delay in interconnection which violates timing specification, a buffer should be inserted. The traditional approach selects the spare cell marked in red color. The ECO wire length of the traditional method which the bounding box contains the input terminal, output terminals and the spare cells, is 4520. In contrast, our approach selected the spare cell marked in black color to reduce the ECO wire length. The wire length covering the ECO node and the spare cell was significantly reduced to 3880. In an observation of Figure 6, we find that the ILP formulations were able to reduce the results of traditional methods. For this reason, how to formulate the ECO problem and solve them by using the ILP formulation motivates us to study the ECO problem.

3. Problem Definition

Based on the previous discussion, the problem of the engineering change order can be formulated as the ILP formulations. Assignment of spare cells with minimal wire length is the objective of the paper. The novel wire insertion-based selector is proposed to select the spare cell with the optimal wire length. In brief, the ECO problem is defined as follows:

Given the original netlist, including a set of 2-pin nets with terminals $\{t_1, t_2, \ldots, t_q\}$ with a source t_0, a set of single-type spare cells $\{s_1, s_2, \ldots, s_m\}$, and a set of engineering change order $\{eco_1, eco_2, \ldots, eco_n\}$ with the corresponding virtual nodes $\{v_1, v_2, \ldots, v_n\}$ to insert buffers, the objective of the ECO problem of aims to select the spare cell assignment with the optimal length.

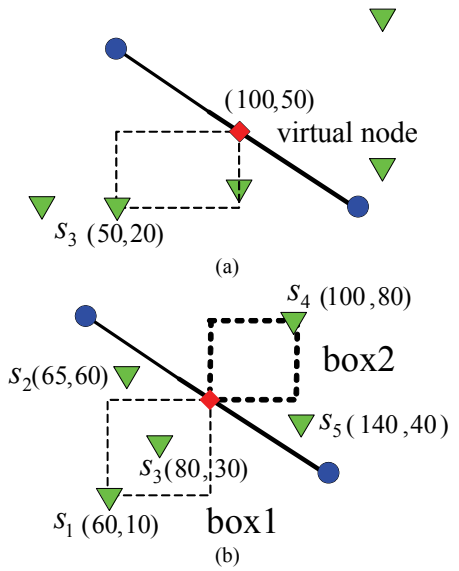


Fig. 5 Spare cell selection by using the insertion-based approach: (a) one spare cell (k=1), (b) two spare cells (k=2)

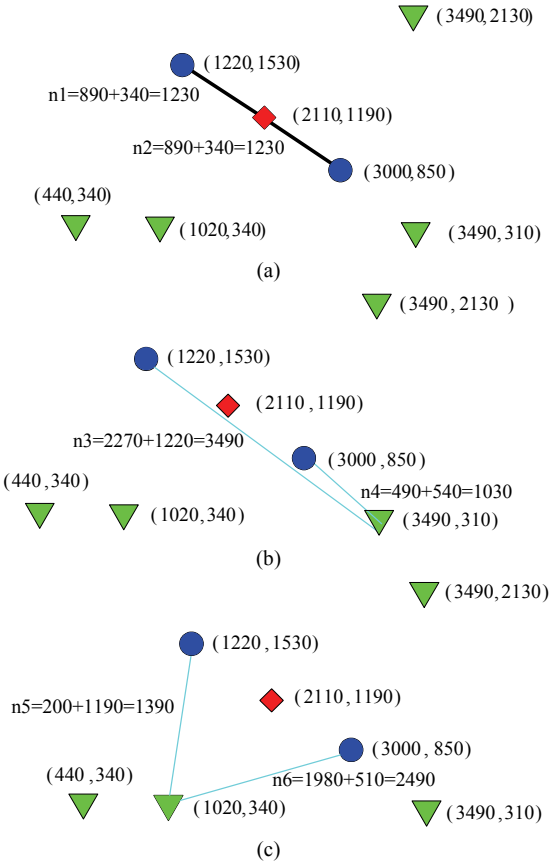


Fig. 6 Superiority of ILP-based approach: (a) original circuit with timing violations, (b) approach in [16], (c) our approach

IV. ECO-DRIVEN ILP FORMULATION

In the section, the integer linear programming based approach is utilized to formulate the ECO problem. Figure 7 illustrates the design flow of our approach.

In ILP formulation, the symbol $b(i, j)$ denotes the binary variable representing the ECO *i* being assigned by the spare cell s_j. If the binary variable $b(i, j)$=1, the spare cell s_j is assigned to ECO eco_i; otherwise, the binary variable $b(i, j)$ is set to be zero. There are *n* ECO in the circuit with corresponding *n* virtual nodes and *m* spare cells which result in the relations of $1 \le i \le n$ and $1 \le j \le m$, respectively. The next sub-section will propose the objective of the paper, capacity of each ECO in the product specification, the resource capacity of each spare cells, and half-meter wire length estimation method.

1. Formulation

The objective is to minimize total wire length due to ECO. In the paper, total length is computed as follows,

$$\text{Min} \quad all\_len\_eco \tag{3}$$

Subject to

For each spare cell s_j,

$$b(1, j) + b(2, j) + \ldots + b(n, j) \le 1 \tag{4}$$

For each ECO eco_i,

$$b(i,1) + b(i,2) + \ldots + b(i,m) = 1 \tag{5}$$

For each distance from virtual node v_i to spare cell s_j,

$$d(i, j) = |y_i - y_j| + |x_i - x_j| \tag{6}$$

For each wire length of eco_i,

$$len\_eco(i) = \sum_{j=1}^{m} d(i, j) \times b(i, j) \tag{7}$$

For the given netlist with *n* ECOs,

$$all\_len\_eco = \sum_{i=1}^{n} len\_eco(i) \tag{8}$$

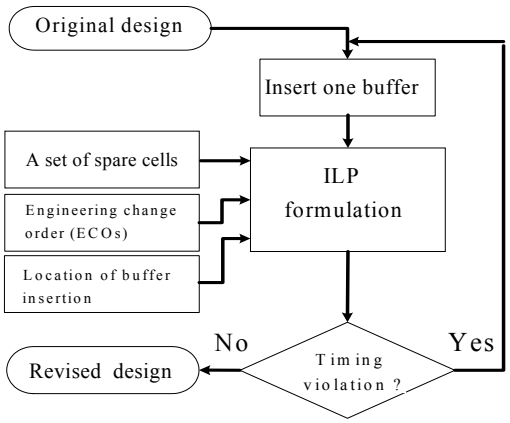


Fig. 7 The design flow

Formula (4) denotes the resource constraint that one spare cell only is assigned at most one ECO to fix timing violations. Formula (5) describes the demand constraint that one ECO can only select one spare cell. Formula (6) states that HPWL-based approach is used to calculate the wire length between the virtual node v_i and the spare cell s_j. Formula (7) states wire length constraint that total wire length is the sum of all distances from virtual node v_i to spare cell s_j. Formula (8) denotes that total wire length of a given netlist with *n* ECOs.

2. An Example

We take the example in Figure 8 to illustrate the spare cell selections problem. We assume that there are 16 spare cells that meet user-defined 3 ECOs issues. To fix timing violation, each ECO is assigned to only one spare cell and the wire length of each ECO is computed by formula (7). Figure 8 gives us the results of spare cell selection for each ECO. For the given ECO resources and the assigned three problems, the objective here is to minimize total length.

Due to the limitation for this paper, we are not able list and explain all constraints of the ILP formulations for ECO problem. In the following section, we will explain all formulas by using an example depicted in Figure 8.

In, formula (4), each spare cell is assigned with only one ECO. For spare cell 2, we have the following,

$$b(1,2) + b(2,2) + b(3,2) \le 1 \tag{9}$$

Similarly, for spare cell 16, we have the following.

$$b(1,16) + b(2,16) + b(3,16) \le 1 \tag{10}$$

In formula (5), only one ECO can choose one spare cell at a time to meet timing violation. For ECO 1, we have the following,

$$b(1,1) + b(1,2) + \ldots + b(1,16) = 1 \tag{11}$$

Similarly, for ECO 3, we have the description of

$$b(3,1) + b(3,2) + \ldots + b(3,16) = 1 \tag{12}$$

In formula (6), each eco_i contains a corresponding virtual node v_i and the wire length from virtual node v_i to spare cell s_j is calculated. The pre-computed wire lengths are generated by the C++ programming, and we integrated the following data, such as $d(1,1) = 4520$, $d(1,2) = 3910$, $d(1,3) = 5690$, … and $d(3,16) = 5165$, into the ILP formulation.

In formula (7), each virtual node v_i has *m* distances to *m* spare cells. Combining the pre-computed wire length described in formula (6) and the binary variables $b(i, j)$, the wire length of each eco_i is calculated. If the pre-computed distance is not selected, the binary variable is set to be "0". Only the distances which their binary variables are set to be "1" are selected. For ECO 1, we obtained the results as follows,

$$\begin{aligned} len\_eco(1) = {} & b(1,1) \times 4520 + \\ & b(1,2) \times 3910 + \ldots + b(1,16) \times 11420 \end{aligned} \tag{13}$$

Similarly, for ECOs 2 and 3 we have the following,

$$len\_eco(2) = b(2,1)\times 5070 + b(2,2)\times 5680 + \ldots + b(2,16)\times 11210 \quad (14)$$

$$len\_eco(3) = b(3,1)\times 8945 + b(3,2)\times 9555 + \ldots + b(3,16)\times 5165 \quad (15)$$

In formula (8), all wire length of n ECOs are considered. In this example, total wire length is the summations of all wire length for 3 ECOs. We have the total wire length described in the following:

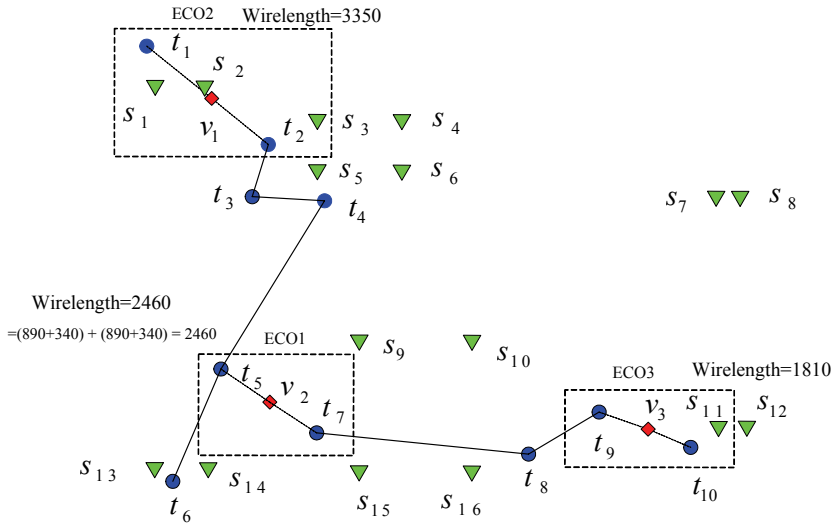
$$all\_len\_eco = \sum_{i=1}^{3} len\_eco(i) \quad (16)$$

After solving the ILP formulations, we obtain the results that $b(1,1)=1$, $b(2,3)=1$ and $b(3,5)=1$, and the other binary values are 0. The results imply that the spare cell assignment with the optimal length for ECOs 1, 2 and 3 are assigned by spare cells 1, 3 and 5, respectively. Figure 8 is the reconnected results. The original wire length is 21520 in Figure 8(a). Coordinates of all nodes are shown in Figure 8(b). The spare cell assignment of the traditional method is shown in Figure 8(c) with the wire length of 24140. The spare cell assignment of our ILP-based approach is shown in Figure 8(d) with wire length of 23500. In short, the proposed approach obtains the spare cell assignment with the optimal length. From the ILP formulation, all constraints for spare cell selection are linear constraints. Hence, the space cell assignment with optimal wire length could be obtained.

V. EXPERIMENTAL RESULTS

The experiments were conducted on an Intel core dual machine with 2GB memory. We apply the linear/nonlinear programming solver, LINGO 11 to solve the associated nonlinear problem for spare cell assignment with wire length minimization. Some benchmarks are applied to verify the performance of the ILP formulations. To simplify, the symbol r is set to be 0.5; i.e. the location of buffer insertion for the dedicated interconnection of performing ECO is the middle node. The proposed ILP-based approach is independent to the parameter (r). The objective is to minimize the total wire length of all ECOs with the resource of spare cells. Due to the lack of the standard ECO benchmarks, the circuits were generated by reference to information of the papers [16]. Each benchmark contains the netlist with connection information, a set of spare cells, a set of terminals and a set of ECOs. To simplify, we only consider one type spare cell (buffer) and insert one buffer at a time for each net (k is 1).

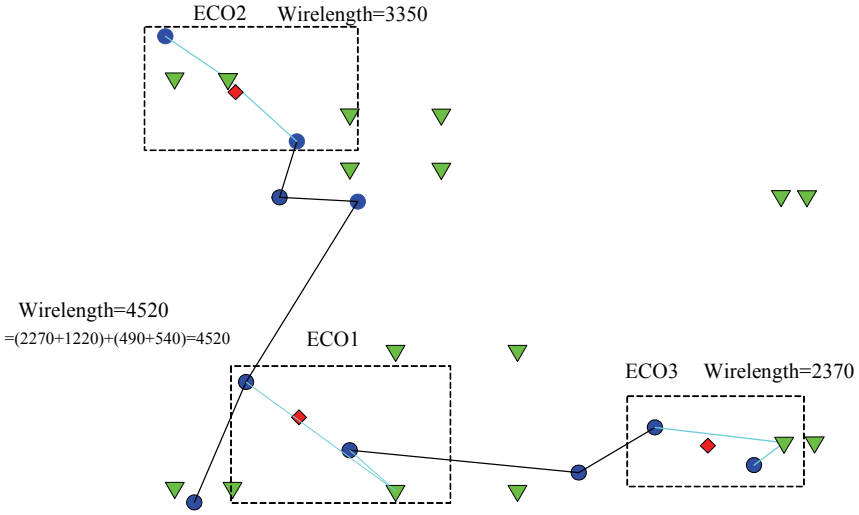
First, the reduction of wire length for all ECOs is investigated. In Table 2, the first, second and third columns denote the name of circuits, the numbers of ECOs and the number of spare cells, respectively. Two bounding box estimation approaches are tested in our experiment. In the first approach, the bounding box covers all input, output terminals, and spare cells. In the second approach, the bounding box only covers virtual nodes and spare cells. "HPWL" and "Insertion" columns in Table 2 denote the wire length results of these different approaches. The column marked with "Results in [16]" denotes the results in [16]. Compared to the first estimation approach, the wire length of the second estimation approach has improved because the concept of eco node only takes into account the middle point of the interconnection and spare cells instead of the region of bounding box covering all input and output terminals, spare cells. In comparison with the results of [16], the wire length is further reduced.



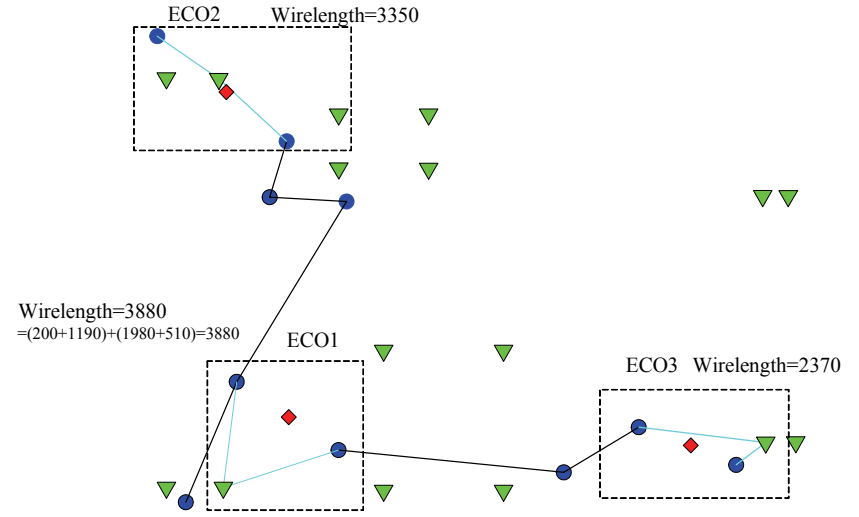
(a) Original length=21520

t_1 (170,6410)	t_2 (2030,4920)
t_3 (1760,4120)	t_4 (2980,4080)
t_5 (1220,1530)	t_6 (580,180)
t_7 (3000,850)	t_8 (6330,610)
t_9 (7480,1160)	t_{10} (8920,790)
v_1 (1100,5665)	v_2 (2110,1190)
v_3 (8200,975)	s_1 (440,5770)
s_2 (1020,5770)	s_3 (2880,5210)
s_4 (4050,5210)	s_5 (2880,4600)
s_6 (4050,4600)	s_7 (9120,4220)
s_8 (9400,4220)	s_9 (3490,2130)
s_{10} (5240,2130)	s_{11} (9120,1240)
s_{12} (9400,1240)	s_{13} (440,340)
s_{14} (1020,340)	s_{15} (3490,310)
s_{16} (5240,310)	

(b) Coordinates of terminals, virtual nodes and spare cells



(c) Length by [16] = 24140



(d) Length by ours = 23500

Fig. 8 Comparisons of wire length

Table 2 Comparison of total wire length and runtime

Circuits	No. of ECOs	No. of spare cells	HPWL		insertion-based		Results in [16]
			Length1 (um)	Time1 (seconds)	Length2 (um)	Time2 (seconds)	
C1	3	20	46530	0	46530	0	46530
C2	5	20	63180	0	63180	0	63320
C3	7	20	85920	0	85920	0	85920
C4	10	20	123580	0	123580	0	123580
C5	10	1000	88007	3	87635	1	88089
C6	20	1000	121321	6	120623	6	121069
C7	20	1600	126179	13	126179	11	126321
C8	20	2000	119544	17	119544	17	119882
C9	50	200	235270	3	235270	2	235390
C10	100	200	397066	3	396858	3	397232
Total	-	-	1406597	-	1405319	-	1407333
Norm	-	-	**1**	**-**	**0.999091**	**-**	**1.000523**

HPWL = the box covering all input, output terminals and spare cells;
Insertion= the box covering virtual nodes and spare cells.
Timer1and Timer2 denote the runtimes of HPWL and insertion-based selector, respectively.

Second, the effects on the runtimes of the novel estimation approaches are compared. The fourth and the fifth columns denote the results of the first estimation approach (HPWL) and the second estimation approach (insertion-based). We observe that the runtimes of insertion-based approaches are short and the approach efficiently obtains the spare cell assignment.

Finally, we compare the spare cell assignment in Figure 8. In each circuit, there are three sub-circuits which are required for modification according the given data in the benchmarks. Three sub-circuits named ECO1, ECO2 and ECO3 are marked in the dotted square line. For the original circuit in Figure 8(a), the heuristic-based approach [16] obtains the results in Figure 8(c) and the wire length is 24140. We observe that rewiring results of ECO2 and ECO3 by the heuristic-based approach [16] and ILP-based approaches are identical. Hence, only the sub-circuit of ECO1 is used for discussion in Figure 6, to demonstrate the difference. In contrast, the ILP-based approach yields the optimal results in Figure 8(d) and the wire length is 23500. Summary, the ILP-based approach gets the optimal wire length of the spare cell assignment. Figure 8 contains three ECOs and the spare cell assignment of ECO2 and ECO3 are identical by using two approaches. Hence, the different spare cell assignment for ECO1 leads to the different results. Therefore, the sub-circuits named ECO1 are discussed in aforementioned Figure 6. The ILP-based approach can obtain the assignment with optimal length.

VI. CONCLUSION

In this paper, we propose the integer linear programming based approach to solve the engineering change order problem. The novel approach, which searches the region covering the middle point of the interconnection and the spare cells, is used to accelerate the runtime of our ILP-based approach. Our experimental results show that our ILP-based approach both efficiently and effectively obtains the spare cell assignment with minimal wire length for ECO.

ACKNOWLEDGEMENTS

This work was supported by National Science Council of Taiwan under Grant No. NSC 99-2221-E-033-065.

REFERENCES

[1] Y. R. Wu, S. Y. Chen, K. Y. Lee, and T. C. Wang, “On using spare cells for functional changes with wirelength consideration,” in *Proc. of Synthesis and System Integration of Mixed Information technologies*, pp. 64-69, 2009.

[2] C. C. Lin, K. C. Chen, and M. M. Sadowska, “Logic synthesis for engineering change,” in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 18, no. 3, pp. 282-292, 1999.

[3] D. Brand, A. Drumm, S. Kundu, and P. Narain, “Incremental synthesis,” in *Proc. of International Conference on Computer-Aided Design*, pp. 14-18, 1994.

[4] K. H. Chang, I. L. Markov, and V. Bertacco, “Reap what you sow: spare cells for post-silicon metal fix,” in *Proc. of Int'l Symposium on Physical Design*, pp. 103-110, 2008.

[5] N. A. Modi and M. Marek-Sadowska, “ECO-map: Technology remapping for post-mask ECO using simulated annealing,” in *Proc. of International Conference on Computer Design*, pp. 652-657, 2008.

[6] A. Kumar, K. Chakrabarty, and C. R. Mohan, “An ECO technique for removing crosstalk violations in clock networks,” in *Proc. of*

VLSI Design, pp. 283-288, 2007.

[7] Y. M. Kuo, Y. T. Chang, S. C. Chang, and M. M. Sadowska, "Engineering change using spare cells with constant insertion," in *Proc. of International Conference on Computer-Aided Design*, pp. 544-547, 2007.

[8] Y. P. Chen, J. W. Fang, and Y. W. Chang, "ECO timing optimization using spare cells," in *Proc. of International Conference on Computer-Aided Design*, pp. 530-535, 2007.

[9] P. Elakkumanan, K. Thyagarajan, K. Prasad, and R. Sridhar, "Optimal vth assignment and buffer insertion for simultaneous leakage and glitch minimization though integer linear programming (ILP)," in *Proc. of IEEE International Midwest Symposium on Circuits and Systems*, pp. 1880-1883, 2005.

[10] S. H. Huang and C. H. Cheng, "An ILP approach to the simultaneous application of operation scheduling and power management," *IEICE Transactions on Fundamentals of Electronics, Communications, and Computer Sciences*, vol. E91-A, no. 1, pp. 375-382, 2008.

[11] A. Davoodi and A. Srivastava, "Effective techniques for the generalized low-power binding problem," *ACM Trans. Design Autom. Electr. Syst.* pp. 52-69, 2006.

[12] S. H. Huang and C. H. Cheng, "Power-management scheduling for peak power minimization," *Journal of Information Science and Engineering (JISE)*, vol. 24, pp. 1647-1668, 2007. (SCI)

[13] R. L. Payne, Cell-Based Integrated Circuit Design Repair Using Gate Array Repair Cells., *US Patent 5959905*, Sep. 1999.

[14] J. Cong, T. Kong, and D. Z. Pan, "Buffer block planning for interconnect-driven floorplanning," *in Proc. of IEEE/ACM International Conference on Computer Aided Design*, pp. 358-363, 1999.

[15] P. Sarkar, V. Sundararaman, and C. K. Koh, "Routability-driven repeater block planning for interconnect-centric floorplanning," *in Proc. of ACM International Symposium on Physical Design*, pp. 50-55, 2000.

[16] Y. K. Yeh, J. H. Hung, Y. S. Tseng, and T. M. Hsieh**,** "Technology remapping for engineering change with wire length consideration," in *Proc.* of *IEEE International Symposium on Circuits and Systems 2010*.