

Integers Sorting Method for Boolean Functions Minimization

Vadym Minzyuk

Abstract - In this paper modification of the most significant digit binary-radix sort has been considered. The method is supplemented by procedure of cardinality counting for subsets, which are derived on steps of sorting. This procedure helps to detect such sets which can be represented by conjuncterms with absorbed low-order bits.

Keywords – sorting, boolean functions, minimization.

I. INTRODUCTION

Boolean function minimization is a part of combinational logic circuits synthesis. On a number of occasions minimization problem requires minterms list sorting. [1]

II. PROBLEM DEFINITION

In this paper modification of most significant digit binary-radix sort has been considered. Let elements of linear list B are n -digit nonnegative binary integers, $d(j, k)$ – digit in j position of binary number k . It is necessary to sort list with m elements.

In terms of logic algebra given list B is a binary image of n -variables Boolean function, and elements of list B are minterms images, where each binary digit is an image of certain Boolean variable.

III. BIT SORTING WITH GLUING

By every bit (beginning from the most significant) let us sort elements set to two subsets. The first subset contains all elements with “0” in the sorting bit. Another subset contains all elements with “1” in that bit. These subsets will be considered separately in process of sorting by other bits.

Simultaneously with sorting we have to count the number of elements in each subset (subsets cardinality). If subset cardinality is equal 2^j , where j — sorting bit number, then all conjuncterms of this subset are gluing by less significant bits. Then this subset can be replaced by conjuncterm, where all left bits (after j -position) are absorbed. Such conjuncterms already take their places in the list and will be skipped on the following sorting by less significant bits.

After completion of sorting by all bits it is necessary to replace each conjuncterm with absorbed bits by list of numbers from 0 to 2^r-1 , where r — quantity of absorbed bits. And nonabsorbed bits of this conjuncterm must be written on the left of each number from 0 to 2^r-1 as more significant bits.

IV. EXAMPLE 1

Sort set of numbers:

{14, 7, 13, 6, 12, 5, 11, 4, 10, 3, 9, 2, 8, 0, 1}.

Vadym Minzyuk – Lviv Politechnic National University, S. Bandery Str., 12, Lviv, 79646, UKRAINE, E-mail: vvmmin@lp.edu.ua

Solution.

The binary image of the numbers:

{1110, 0111, 1101, 0110, 1100, 0101, 1011, 0100, 1010, 0011, 1001, 0010, 1000, 0000, 0001}.

Sorting by the most significant (third) bit:

{0111, 0110, 0101, 0100, 0011, 0010, 0000, 0001},
{1110, 1101, 1100, 1011, 1010, 1001, 1000}.

The first subset cardinality is equal 2^3 . So this subset is replaced by conjuncterm $\{(0 \text{ - - -})\}$. This conjuncterm already takes its place in the list and will be skipped on the following sorting by less significant bits.

$\{(0 \text{ - - -})\}$, {1110, 1101, 1100, 1011, 1010, 1001, 1000}.

Sorting by second bit:

$\{(0 \text{ - - -})\}$, {1011, 1010, 1001, 1000}, {1110, 1101, 1100}
 $\{(0 \text{ - - -})\}$, $\{(10 \text{ - -})\}$, {1110, 1101, 1100}.

Sorting by first bit

$\{(0 \text{ - - -})\}$, $\{(10 \text{ - -})\}$, {1101, 1100}, {1110}.
 $\{(0 \text{ - - -})\}$, $\{(10 \text{ - -})\}$, $\{(110 \text{ -})\}$, {1110}.

Let us replace conjuncterm $\{(0 \text{ - - -})\}$ by ordered set of numbers. As three bits are absorbed, let us write list of numbers from 0 to (2^3-1) . And let us write nonabsorbed bits of conjuncterm $\{(0 \text{ - - -})\}$ (that is “0”) on the left of each numbers from 0 to (2^3-1) :

{0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111}.

After recovering all absorbed bits of each conjuncterm

{0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111},
{1000, 1001, 1010, 1011}, {1100, 1101}, {1110}.

Final result in decimal image

{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14}.

V. CONCLUSION

Some methods of calculating prime implicants of Boolean functions (for example, method bit-by-bit growing up) [1] need preliminary sorting of minterms. Use of bit sorting with gluing make it possible to get some quantity of lesser rank conjuncterms even before minimization procedure. And there is no need to search gluing conjuncterms by the least significant bit in the sequel. Because of all such absorptions were found by sorting.

REFERENCES

- [1] Лузин С.Ю. Асимптотически оптимальный метод получения простых импликант. / Автоматика и вычислительная техника – 2000. – №1. – с.80-84.