Research Note

SINGLE ARRAY BRANCH AND BOUND METHOD

S. Khanmohammadi

Electrical Engineering Department
University of Tabriz
Tabriz, Iran

Abstract One of the major disadvantages of using Branch and Bound algorithm to solve permutation bases problems by computer is the size of main memory required. The Single Array Branch and Bound (SABB) method introduced here aims to overcome this deficiency, using only N memory locations as a vector of size N, and a single cell C with N bits.

چکیده یکی از معایب استفاده از الگوریتم شاخه وحد در حل مسایل جایگشتی حجم حافظه اصلی مورد نیاز در کامپیوتر است. در این مقاله الگوریتم شاخه وحدتك آرایه برای غلبه بر این مشکل معرفی میشود. در این الگوریتم برای یافتن بهترین ترتیب بین !n جایگشت ممکن فقط از n سلول حافظه بصورت يك بردار n و يك سلول ساده C با n بیت استفاده شود.

INTRODUCTION

Real time optimal control by microprocessors or microcomputers is widely developing. Finding the optimal sequence to service N jobs in a single or more stations, such as queuing, traffic, machining and production systems is one of the generally encountered problems in such controls [1 , 2 , 3]. The Branch and Bound algorithm is generally used to solve a variety of such problems [4 , 5]. One of the major disadvantages of this algorithm is the big size of memory needed.

The SABB Method

In the single Array Branch and Bound (SABB) procedure introduced here; the branch and bound tree is constructed in another form. Each vertex contains only one number. Vertexes are divided into different levels. By choosing one vertex at a time from each level, a new job is added to the sequence. Figure 1 represents this type of Branch and Bound tree.

To find all possible permutations, it suffices to follow a path which starts from the root vertex and returns back to it as shown inFigure 1.When a vertex is reached for the first time, it is added to the sequence; and for the second time it is removed. Reaching a leaf means that a new sequence is obtained.

Considering each vertex as an on-off unit, when a vertex is passed for the first time (from its right in Figure 1(a)), it becomes on; and for the second time (from its

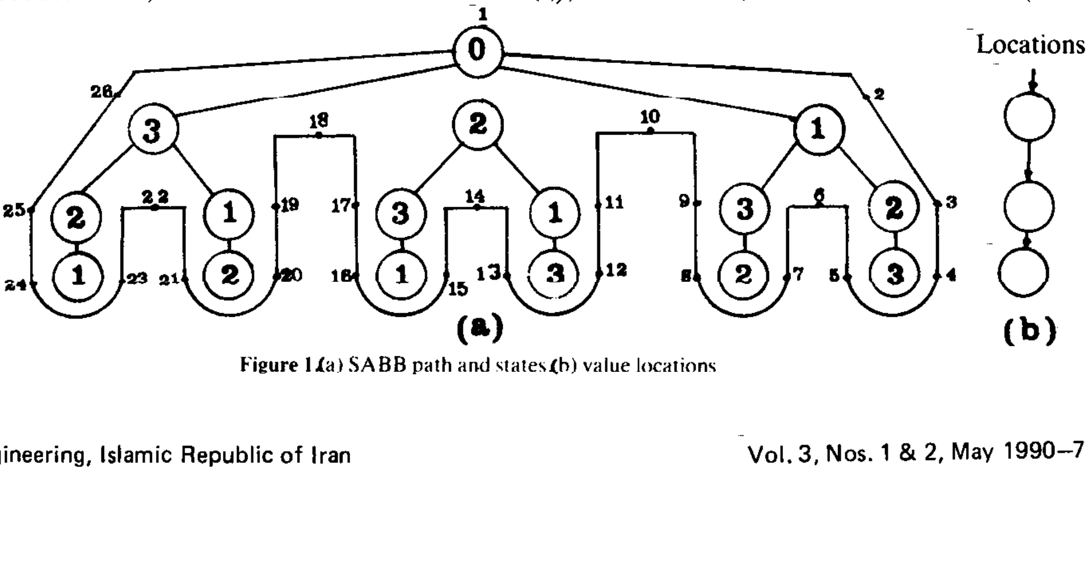

Figure 1.(a) SABB path and states,(b) value locations

left), it becomes off. Only one of the vertexes of each level may be on at a time where its value can be seen in a set of N locations (which is represented in computer as a N dimensional vector $P_{1\times N}$) as shown in Figure 1(b) The states of vertexes are represented by bits of a memory cell in computer. $Bit_j=1$ means that job is in the queue(it is on in Figure 1) and $bit_j=0$ denotes that job j is free(it is off in Figure 1), and can be added to the sequence. This procedure can be written as a computer program to find all possible permutations of the jobs.

ILLUSTRATIVE EXAMPLE

An illustrative example to study 3! possible permutations of 3 jobs assuming that no dropping occurs during the procedure, is shown in Figure 2.

CONCLUSION

The SABB procedure is a software mechanism introduced to overcome the problem of main memory required in real - time optimal control of sequencing systems. The minisize SABB algorithm can be widely used in optimum process controls such as optimum sequences of robot arms movements, operations of uinversal machines, services in traffic systems, etc.

REFERENCES

1. Sequencing to minimize the maximum job cost. CLYDE L.MONMA Opertations research, vol 28, No. 4, July - Agust 1980
2. Stable, Distributed, Real - Time scheduling of Flexible Manufacturing / Assembly / Disassembly Systems. J. R. PERKINS and P. R. KUMAR IEEE Trans. on Automatic Control, vol. 34, No. 2, Feb. 1989.
3. Simulation of air traffic control operation in a zone of convergence. A. BENOIT, S. SWIERSTRA Eurocontrol, division El, doc 812031 - 2, Apr. 1982.
4. Branch and Bound methods for Multi - Item scheduling. DNNIS J. SWEENEY and RICHARD A. MURPHY. Operations research, vol 29, no. 5, Sep. - Oct. 1981.
5. A Branch and Bound algorithm for the symmetric travelling salesman problem based on the 1-tree relaxation. TON VOLGENANT, ROY JONKER European journal of operational res., vol. 9, No. 1, Jan. 1982.

States: Elements of P: Bits of C:	(1) (000) (000)	(2) (100) (100)	(3) (120) (110)	(4) (123) (111)	(5) (120) (110)
(6) (1 3 0) (1 0 1)	(7) (1 3 2) (1 1 1)	(8) (1 3 0) (1 0 1)	(9) (1 0 0) (1 0 0)	(10) (2 0 0) (0 1 0)	(11) (2 1 0) (1 1 0)
(12) (2 1 3) (1 1 1)	(13) (2 1 0) (1 1 0)	(14) (2 3 0) (0 1 1)	(15) (2 3 1) (1 1 1)	(16) (2 3 0) (0 1 1)	(17) (2 0 0) (0 1 0)
(18) (3 0 0) (0 0 1)	(19) (3 1 0) (1 0 1)	(20) (3 1 2) (1 1 1)	(21) (3 1 0) (1 0 1)	(22) (3 2 0) (0 1 1)	(23) (3 2 1) (1 1 1)
(24) (3 2 0) (0 1 1)	(25) (3 0 0) (0 0 1) →	(26) (0 0 0) (0 0 0)			

Figure 2. Illustrative example of SABB procedure fo jobs.