Computer Simulation-based Optimization: Hybrid Branch & Bound and Orthogonal Array based Enumeration Algorithm

Mohamed H. Gadallah

Abstract— **We propose a new modified algorithm for the Branch and Bound method. This method is based on integration of orthogonal arrays and enumeration based techniques. Several observations are given supplemented with simple model solutions to verify our assumptions. The modified algorithm is valid for higher dimensional problems. The algorithm employs several 2-levels orthogonal arrays and the complexity of solutions is always of the order 2^n . For low to medium size problems, full factorial orthogonal arrays are employed. For higher size problems, the user uses fractional orthogonal arrays and the complexity of solution is always of the order 2^{n-k}, $k \le n$. Results show that full size arrays yield optimum solution consistently. When fractional arrays are used, the solution is always incumbent. The number of function evaluations is always low. We show that this conclusion, though true, is rare as the number of continuous solutions resulting from relaxations is always tractable with full size arrays. This method utilizes the fact that the number of continuous solutions from relaxations is lesser than the original problem size. Numerical results show that the proposed hybrid algorithm is able to save 20% ~ 96% of the original computations. Fractional arrays allow fractionation and consequent deviation from the best solution.**

Index Terms— **Approximation, B. & B. Algorithms, optimization, hybrid techniques.**

I. Introduction

Standard optimization methods such as Branch and Bound (B. & B.) are used to deal with problems that are 0-1 binary programming, mixed integer programming (MIP) and general integer programming (IP). B. & B. method is efficient but size dependent. Engineering problem synthesis, analysis and improvements are becoming crucial nowadays. As knowledge advances, problem analysis and synthesis increase in complexity. Accordingly, the synthesis tool should develop at the same pace.

Manuscript received April 18, 2009. Mohamed H. Gadallah is Associate Professor, Department of Operations Research, Institute of Statistical Studies & Research, Cairo University, Cairo, Egypt 12613 (phone: 202-37484624; fax: 202-37482533; e-mail: Mohamed@aucegypt.edu).

Most realistic problems are complex to solve with existing optimization techniques. We admit that there are complex realistic engineering problems, commonly known as NP-hard. This hardship is due to the problem size, combinatorial nature, storage capacity, etc. Part of hardship, of course, is due to the synthesis phase "how to model the optimization problem?" An algorithm is developed that integrates the B. & B. method with orthogonal arrays and enumeration techniques. The search domain is modeled using orthogonal arrays (full or fractional) and enumeration techniques. We show that there is equivalence between both the space used and the resulting problem solution. The proposed algorithm can determine the search size and accordingly the cost of solution. For the first time, the algorithm gives the user the flexibility to choose the solution quality and corresponding cost. In case a high quality solution is required, the modeller should be ready to compromise time and effort. Several situations do not require quality solutions; accordingly, the user can resort to fractional arrays to model the problem. Integration of B. and B. algorithm, orthogonal arrays and enumeration techniques is novel to our knowledge and the engineering community should welcome such means of hybridization as long as they offer solution to hard to solve problems.

Realistic engineering applications are size dependent. The engineering community is either considering the option of over simplification of the original problem or resort to hybrid methods. Over simplification yields trivial solutions. The B.& B. algorithm is the search engine, the orthogonal arrays are the search domains and enumeration techniques are the possibilities of all model formulations. Due to space limitation, past studies are not included.

II. Nomenclature

2^n	2 levels factorial design, n=# of variables.
L_4OA	4 experiments, 2-levels orthogonal arrays.
X_{ij}	Decision variables from i to j.
f	Objective function.
X^c	Continuous form of decision variable X .
$L_{64}OA$	64 experiments, 2-levels orthogonal arrays.
$P_1,\ldots,P_{64}$	64 sub-models.
f_{min}	Minimum objective value.
f_{max}	Maximum objective value.
P_{11}	A diverse of sub-model P_1
P_{12}	Another diverse of sub-model P_1

ISBN: 978-988-17012-5-1

Y_{ij}	Decision variables from i to j.
L_2OA	2-levels orthogonal array (single variable).
P_{121}	A diverse of sub-model P_{12}
P_{122}	Another diverse of sub-model P_{12}
P_0	Original model (after relaxation).
$L_{16}OA$	16 experiments, 2-levels orthogonal arrays.
$L_{32}OA$	32 experiments, 2-levels orthogonal arrays.
2^{k-k_1}	Fractional factorial, 2 levels arrays $(K \geq K_1)$.
X^*	Optimum solution of decision variable.
$L_{65,536}OA$	65,536 exp., 2-levels orthogonal arrays.
$L_{2048}OA$	2048 experiments, 2-levels orthogonal arrays.
$L_{\infty}OA$	Very huge, 2-levels orthogonal arrays (2^{44}).

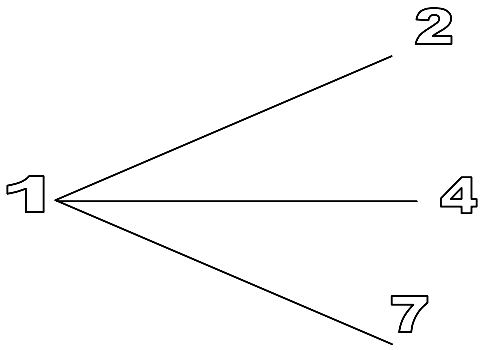
III. METHODOLOGY

Table 1 shows the array sizes used for different number of variables. The full and fractional arrays are given with the probability of finding the best solution (optimum). For instance, an L4OA is used to model 2 variables with a probability of 100% to find the optimum. As the number of variables increase to 3, the array size changes from L4OA to L8OA. The analyst has an option of using : a) an L8OA and 100% probability of getting the best solution or b) an L4OA and 50% probability of getting the best solution. Equivalently, L16OA is used to model 4 variables with a probability of 100% to find the best solution. Similarly, the analyst may decide to fractionate by using L8OA (and L4OA). In this case, the probability of finding the best solution is 50% for L8OA. As the number of variables increase to 9, the array size becomes L512OA (for the full array), L256OA (for half of the array), L128OA (for the quarter of array), L64OA (for the 1/8 of array) and L32OA (for the 1/16 of array) respectively. When the user uses the full size array, the probability of getting the best solution is 100%. Once he decides to fractionate, the probability of getting the best solution is equivalent to the fraction value. For instance, when L256OA is used to model a 9-variable problem, the probability becomes 25%. Ideally the modeller will try to employ the full size arrays to enjoy the best solutions. Full size arrays are tractable, easy to code and understand for low size problems ($n \leq 5 \sim 6$ variables). For a medium size problem with $n \geq 15$ variables, the size becomes $2^{15} = 32,768$ and only then, the modeller will start to realize the benefit of fractionation.

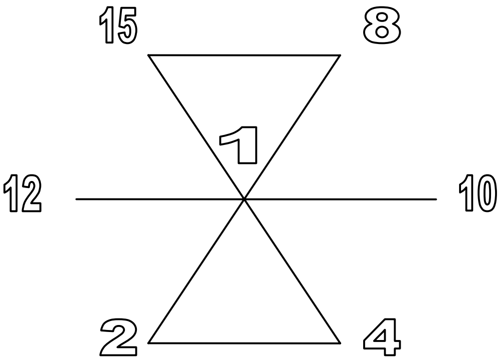
Different search graphs assign continuous variables resulting from constraint relaxation to different arrays columns. Figure 1 gives different search graphs for L8OA, L16OA and L32OA respectively. For instance, when L8OA is used, variables can be assigned to columns 1, 2 and 4 (for a full array) or 1, 2, 4 and 7 (for a half array). Similarly, when L16OA is used, variables can be assigned to columns 1, 2, 4 and 8 (for a full array) or 1,2,4,8 and 15 (for a half array) or 1,2,4,8,15 and 12 (for a quarter array) respectively. Similar assignments can be followed once L32OA is used.

Table 1: Array Types, Maximum Number of Modeled Variables for Full & Fractional Arrays.

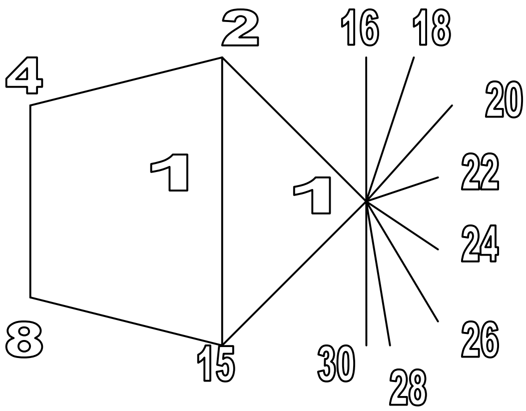
	Array Size/ Type	Probability of finding optimum = 100%	Probability of finding optimum < 100%	Fraction-al factorial Arrays
		# of possible variables	# of possible variables as fractional size array	
Full factorial Arrays	L4OA	2	-	
	L8OA	3	$4 \equiv 1/2$ - FFE , $5 \equiv 1/4$ - FFE	
	L16OA	4	$5 \equiv 1/2$ - FFE , $6 \equiv 1/4$ - FFE	
	L32OA	5	$6 \equiv 1/2$ - FFE , $7 \equiv 1/4$ - FFE , $8 \equiv 1/8$ - FFE	
	L64OA	6	$7 \equiv 1/2$ - FFE , $8 \equiv 1/4$ - FFE , $9 \equiv 1/8$ - FFE	
	L128OA	7	$8 \equiv 1/2$ - FFE , $9 \equiv 1/4$ - FFE , $10 \equiv 1/8$ - FFE	
	L256OA	8	$9 \equiv 1/2$ - FFE , $10 \equiv 1/4$ - FFE , $11 \equiv 1/8$ - FFE	
	L512OA	9	$10 \equiv 1/2$ - FFE , $11 \equiv 1/4$ - FFE , $12 \equiv 1/8$ - FFE	



L8OA search graph



L16OA search graph



L32OA search graph

Fig. 1: L8OA, L16OA and L32OA and corresponding search graphs.

Case Study 1: MIMCK Problem [5].
This is a model with 21 continuous and 9 binary 0,1 variables and linear objective and 4 inequality constraints. The model is given as:

Maximize

$$f = 3X_{11} + 5X_{12} + 8X_{13} + 6X_{14} + 5X_{15} + 6X_{16} + 7X_{17} + 2X_{21} + 6X_{22} + 4X_{23} + 6X_{24} + 3X_{25} + 6X_{26} + 7X_{27} + 2X_{31} + 4X_{32} + 4X_{33} + 8X_{34} + 3X_{35} + 5X_{36} + 9X_{37} + 3Y_{11} + 6Y_{12} + 3Y_{13} + Y_{21} + 7Y_{22} + 6Y_{23} + 4Y_{31} + 3Y_{32} + 5Y_{33}$$

s.t.:

$$g_1: X_{11} + 2X_{12} + 4X_{13} + 3X_{14} + 6X_{15} + 5X_{16} + 4X_{17} + 5X_{21} + 2X_{22} + X_{23} + 3X_{24} + 6X_{25} + 5X_{26} + 3X_{27} + 5X_{31} + 2X_{32} + 4X_{33} + 2X_{34} + 6X_{35} + X_{36} + 3X_{37} + 7Y_{11} + 5Y_{12} + 4Y_{13} + 2Y_{21} + 6Y_{22} + 4Y_{23} + 7Y_{31} + 6Y_{32} + Y_{33} \le 40$$

$$g_2: X_{11} + X_{12} + X_{13} + X_{14} + X_{15} + X_{16} + X_{17} \le 1 ;$$

$$g_3: X_{21} + X_{22} + X_{23} + X_{24} + X_{25} + X_{26} + X_{27} \le 1 ;$$

$$g_4: X_{31} + X_{32} + X_{33} + X_{34} + X_{35} + X_{36} + X_{37} \le 1 ;$$

$$g_5: X_{ki} \ge 0, \ k = 1,2,3, \ i = 1,2, \cdots,7$$

$$g_6: Y_{kj} \in \{0,1\}, \ k = 1,2,3, \ j = 1,2,3$$

When the model is synthesized as continuous, the best maximum reached is $56.50.

$X^c = (0,0,1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1)$ and

$Y^c = (0,1,1,0,1,1,1,0.5,1)$ and $f_{max} = 56.50$. Two sub-models are formulated and shown in Table 2.

Table 2: Different Sub-models, corresponding Y conditions and f_{max} values.

Prob	Y_{32}	f_{max}	Solution	
P_{11}	≤ 0	56.43	$Y_{11} = 0.14$	$P_{111}: Y_{11} \le 0$, f_{max} =56, optimum
				$P_{112}: Y_{11} \ge 1$, f_{max} =55.71
P_{12}	≥ 1	56.29	$Y_{31} = 0.57$	$P_{121}: Y_{31} \le 0$, f_{max} =55.86
				$P_{122}: Y_{31} \ge 1$, f_{max} =55.75

The 2 sub-models are given below:

P_{11} : Maximize f Subject to: $g_1, \cdots g_4$;
$0 \le Y_{11}, Y_{12} \ldots Y_{33} \le 1$; $Y_{32} \le 0$

P_{11} yields f_{max} =56.43 and $Y_{11} = 0.14$. P_{11} will produce 2 further sub-models: P_{111} & P_{112}

ISBN: 978-988-17012-5-1

P_{111} : Maximize f Subject to: g_1 , ⋯ g_4 ; $0 \leq Y_{11}, Y_{12} \ldots Y_{33} \leq 1$; $Y_{32} \leq 0$, $Y_{11} \leq 0$

P_{111} yields f_{max} =56 and this is the optimum value.

P_{112} : Maximize f Subject to: g_1 , ⋯ g_4 ; $0 \leq Y_{11}, Y_{12} \ldots Y_{33} \leq 1$; $Y_{32} \leq 0$, $Y_{11} \geq 1$

P_{112} yields f_{max} =55.71.

P_{12} : Maximize f Subject to: g_1 , ⋯ g_4 $0 \leq Y_{11}, Y_{12} \ldots Y_{33} \leq 1$; $Y_{32} \geq 1$

P_{12} yields f_{max} =56.29 and $Y_{31} = 0.57$. P_{12} will produce 2 further sub-models: P_{121} & P_{122}

P_{121} : Maximize f Subject to: g_1 , ⋯ g_4 ; $0 \leq Y_{11}, Y_{12} \ldots Y_{33} \leq 1$; $Y_{32} \geq 1$, $Y_{31} \leq 0$

P_{121} yields f_{max} =55.86.

P_{122} : Maximize f Subject to: g_1 , ⋯ g_4 ; $0 \leq Y_{11}, Y_{12} \ldots Y_{33} \leq 1$; $Y_{32} \geq 1$, $Y_{31} \geq 1$

P_{122} yields f_{max} =55.75.

Case Study 2: Limitations to Fractional Based Search

We present a case study with 30 binary integer program. We show how the fractional search fails to obtain the best solution because of problem size. This model is known as Maximum Coverage EMS Model [17] and is given by:

Minimize

$$f = 5.2Y_1 + 4.4Y_2 + 7.1Y_3 + 9.0Y_4 + 6.1Y_5 + 5.7Y_6 + 10Y_7 + 12.2Y_8 + 7.6Y_9 + 20.3Y_{10} + 30.4Y_{11} + 30.9Y_{12} + 12Y_{13} + 9.3Y_{14} + 15.5Y_{15} + 25.6Y_{16} + 11Y_{17} + 5.3Y_{18} + 7.9Y_{19} + 9.9Y_{20}$$

s.t.: $g_1 : X_2 + Y_1 \geq 1$ $g_2 : X_1 + X_2 + Y_2 \geq 1$;

$g_3 : X_1 + X_3 + Y_3 \geq 1$; $g_4 : X_3 + Y_4 \geq 1$;

$g_5 : X_3 + Y_5 \geq 1$; $g_6 : X_2 + Y_6 \geq 1$;

$g_7 : X_2 + X_4 + Y_7 \geq 1$; $g_8 : X_3 + X_4 + Y_8 \geq 1$;

$g_9 : X_8 + Y_9 \geq 1$; $g_{10} : X_4 + X_6 + Y_{10} \geq 1$;

$g_{11} : X_4 + X_5 + Y_{11} \geq 1$;

$g_{12} : X_4 + X_5 + X_6 + Y_{12} \geq 1$;

$g_{13} : X_4 + X_5 + X_7 + Y_{13} \geq 1$;

$g_{14} : X_8 + X_9 + Y_{14} \geq 1$;

$g_{15} : X_6 + X_9 + Y_{15} \geq 1$;

$g_{16} : X_5 + X_6 + Y_{16} \geq 1$;

$g_{17} : X_5 + X_7 + X_{10} + Y_{17} \geq 1$;

$g_{18} : X_8 + X_9 + Y_{18} \geq 1$; g_{19} : $X_9 + X_{10} + Y_{19} \geq 1$;

$g_{20} : X_{10} + Y_{20} \geq 1$; g_{21} : $\sum_{j=1}^{10} X_j \leq 4$; $X_1 \cdots X_{10} = 0,1$ $Y_1 \cdots Y_{20} = 0,1$

Relaxation of P_0 model yields $Y_1 = Y_2 = Y_3 = Y_4 = Y_5 = Y_6 = Y_9 = Y_{20} = 0.5$ and $X_2 = X_3 = X_4 = X_5 = X_6 = X_8 = X_9 = X_{10} = 0.5$ This means that the continuous branching variables = 16 (out of 30 original variable). The array size becomes $2^{16} = 65{,}536 \Leftrightarrow L65{,}536OA$. Certainly this is very prohibitive array size. The full size array is very hard to develop and the assignment of variables to L65,536OA is cumbersome. This means that full size arrays, search graphs, quality and cost of solution are very restrictive. Fractional arrays and corresponding approximate solutions are considered. Accordingly, an L32OA is used to model the 16 branching variables. We could not use L16OA as it has 15 degrees of freedom (can host only 15 variables). This model is made of 30 variables and 21 constraints. Constraints are of equality/ inequality nature. Pure enumeration method yields a hard-to-solve problem. With the L32OA, 32 sub-models are formulated and solutions are recorded. Only 5 solution results and 27 models returned "a no-feasible solution". The solutions are summarized in Table 3 for brevity. For instance, Trial # 14 has $Y_2 = Y_3 = Y_{20} = X_2 = 2$, (2nd level, =1) and $Y_6 = X_4 = 1$, (1st level, = 0). In other words, six further variables can be fixed and the 16 variable relaxed model becomes 10 instead of 16. Further insight reveals no function value difference between Trial 14 and 32. Hence, setting $Y_1, Y_4, Y_5, X_5, X_8, X_9$ at 1st level or 2nd level has no impact on the objective function. The same can be stated for Y_9, X_3, X_6, X_{10} respectively. Another look at trials # 4 and # 25 would conclude that different variable settings would result in almost similar function values $f_{min} = 67.3$ for trial # 4 vs. $f_{min} = 68.7$ for trial # 25. Y_3, Y_4, X_6, X_9 can be set at the 1st level and Y_6, Y_9, X_3, X_5 can be set at the high level. This means the 16 variable model can be reduced to an 8 variable model. Besides, Y_1, Y_2, X_8, X_{10} set at either the low or high levels would not affect the objective function. A similar conclusion can be stated for Y_5, Y_{20}, X_2, X_4 respectively. Four sets of experiments are given next.

Experiment 1: Effect of Y_1, Y_2 on solution

Y_1, Y_2 are modelled using L4OA. 4 sub-models are formulated and solved. All sub-models yielded "no-feasible solution".

Experiment 2: Effect of Y_1, Y_2, Y_3, Y_4 on solution

Y_3, Y_4 are modelled using L4OA with $Y_1, Y_2 \leq 0$. 4 sub-models are formulated and solved. All sub-models yielded "no-feasible solution".

Experiment 3: Effect of $Y_1, Y_2, Y_3, Y_4, Y_5, Y_6$ on solution

Y_5, Y_6 are modelled using L4OA with $Y_1, Y_2 \leq 0, Y_3, Y_4 \leq 0$. 4 sub-models are formulated and solved. All sub-models yielded "no-feasible solution".

Table 3: Maximum Coverage Problem using L32OA

Trial	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_9	Y_{20}	X_2	X_3	X_4	X_5	X_6	X_8	X_9	X_{10}	f_{min}
4	1	1	1	1	2	2	2	2	2	2	2	2	1	1	1	1	$f_{min} = 67.3$
8	1	1	2	2	2	2	2	2	2	2	1	1	1	1	2	2	$f_{min} = 152.6$
14	1	2	2	1	1	1	2	2	2	2	1	1	2	1	1	2	$f_{min} = 74$
25	2	2	1	1	1	2	2	1	1	2	1	2	1	2	1	2	$f_{min} = 68.7$
32	2	2	2	2	2	1	1	2	2	1	1	2	1	2	2	1	$f_{min} = 74.2$

$1 \equiv \leq 0$, $2 \equiv \geq 1$

Experiment 4: Effect of $Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_9, Y_{20}$ on solution

Y_9, Y_{20} are modelled using L4OA with $Y_1, Y_2 \leq 0, Y_3, Y_4 \leq 0, Y_9, Y_{20} \leq 0$. 4 sub-models are formulated and solved. All sub-models yielded "no-feasible solution".

With these preliminary results, we can assert that L32OA is not enough to model the problem and other larger size array has to be used. Besides, we can conclude that the model has no feasible solution with $Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_9, Y_{20} \leq 0$.

Case Study 3: Minimum Coverage Problem [17].

This is a problem with 10 binary 0,1 variables and linear objective and constraints. The model is given as:

Minimize $f = \sum_{j=1}^{10} X_j$

s.t.: $g_1 : X_2 \geq 1$; $g_2 : X_1 + X_2 \geq 1$; $g_3 : X_1 + X_3 \geq 1$; $g_4 : X_3 \geq 1$; $g_5 : X_2 \geq 1$; $g_6 : X_2 + X_4 \geq 1$; $g_7 : X_3 + X_4 \geq 1$; $g_8 : X_2 \geq 1$; $g_9 : X_6 + X_4 \geq 1$; $g_{10} : X_4 + X_5 \geq 1$; $g_{11} : X_4 + X_5 + X_6 \geq 1$; $g_{12} : X_4 + X_5 + X_7 \geq 1$; $g_{13} : X_8 + X_9 \geq 1$; $g_{14} : X_6 + X_9 \geq 1$; $g_{15} : X_6 + X_5 \geq 1$; $g_{16} : X_4 + X_5 \geq 1$; $g_{17} : X_7 + X_5 + X_{10} \geq 1$; $g_{18} : X_8 + X_9 \geq 1$; $g_{19} : X_{10} + X_9 \geq 1$; $g_{20} : X_{10} \geq 1$; $X_{ij} = 0,1$;

The model is synthesized as 0-1. The best minimum reached is 6. Four idealizations are given in Table 4 using $L_{12}OA, L_{16}OA, L_{32}OA, L_{36}OA$ respectively.

Table 4: Different Fractional Arrays & model solutions.

Array Employed	**L12OA**	**L16OA**	**L32OA**	**L36OA**
% of fractionation	1.1718%	1.5625%	3.125%	3.5156%
Min. Objective	6	No Feasible Solution	9	6
Max. Objective	No Feasible Solutions		No Feasible Solutions	No Feasible Solutions
Optimum settings	{0,1,1,0,1,1,0,1,0,1}	N/A	{1,1,1,1,0,1,1,1,1,1}	{0,1,1,0,1,1,0,1,0,1}
Non Standard Array	√	√	√	√

IV RESULTS & DISCUSSION

Table 5 gives a summary of all problems examined. The original model is described in terms of # of variables, # of constraints, nature of problems and solution obtained. The maximum number of continuous variables is also given. The maximum number of continuous variables is 16 (out of 30) for the Max_MCEMS_1. This is the only exception out of the problems tested. In the reduced model, the sizes of different arrays used to model the resulting space range from L2OA ~ L64OA. The employed orthogonal array sizes are 2-levels arrays of acceptable sizes and the cost is always 2^n. The cost of the method employed is always affordable. Compared with the regular B. and B. algorithm, the proposed method has a positive effect in reducing the initial problem size by 20% ~ 96.6%. Figure 2 gives a summary of tested cases in terms of # of variables (initially and after application of hybrid algorithm) and the % of size reduction achieved.

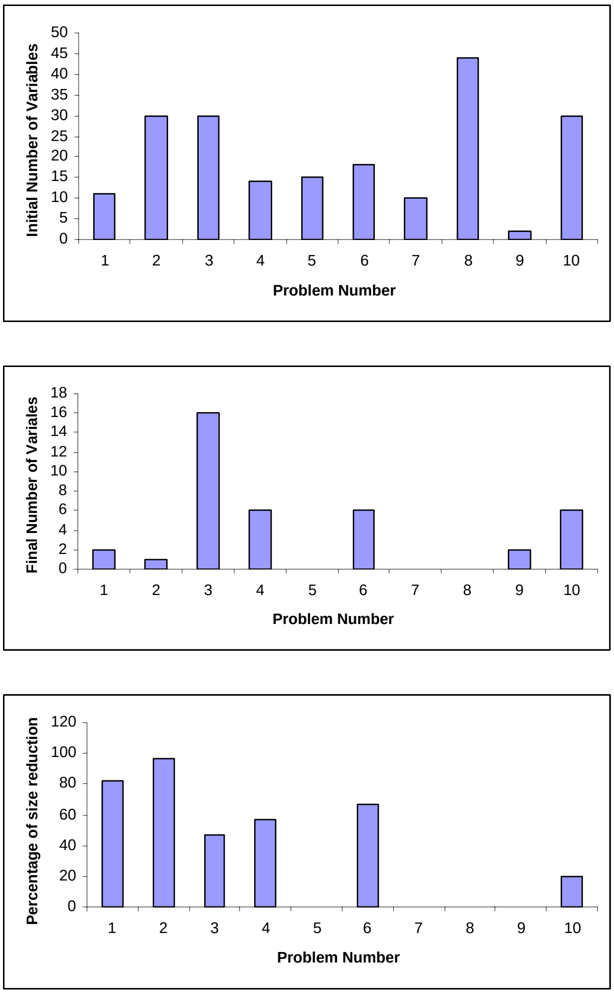


Fig. 2: Summary of test cases.

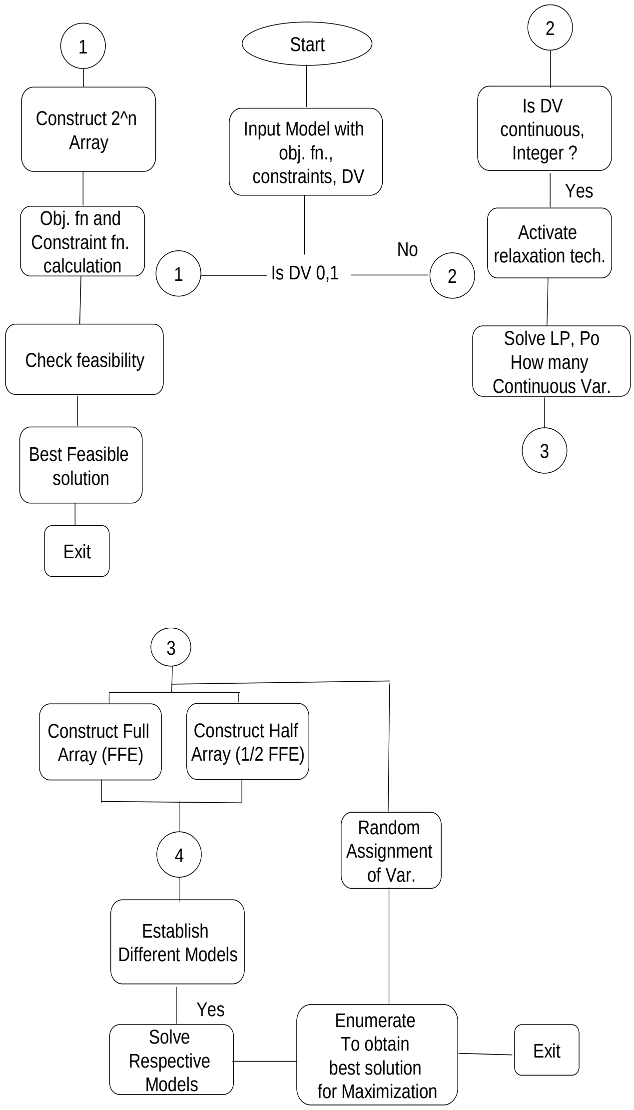


Fig. 3: B. & B. Algorithm via Orthogonal Array Based Enumeration Techniques

Table 6 gives description of different arrays. Only 2-level arrays are used, some are standard such as L4OA, L8OA, L16OA, L32OA and L64OA. Others are not standard such as L128OA, L256OA, L512OA, L1024OA and L2048OA. The development of search graphs for non standard arrays and high number of variables is a novel area of research. This research only utilizes existing search graphs for standard low size arrays. The knowledge of search graphs limits the use of non standard arrays, although can be used. These standard and non standard arrays can model 1~11 variables (for standard arrays) and 3~15 (for non standard arrays). Low size arrays are less expensive than large size arrays. The full size arrays require expensive # of function evaluations and CPU time. The results obtained are high quality unique solutions. When fractional arrays are used, certainly more variables can be modelled at the expense of non-unique solutions. The solution expense increases with the number of variables and size of array. The algorithm developed is given in figure 3.

V. CONCLUSIONS

- NP-hard problems can be solved via the proposed method in a cost effective manner.
- The method proved applicable to LP problems. Extension to quadratic and nonlinear problems requires linearization techniques about a point.
- Orthogonal arrays used, so far, are 2^k of acceptable size. Further larger size problems need to check the solution expense of arrays employed.
- The proposed method needs a comparative analysis with other methods vs. the size of different problems. This will help place our method relative to others with respect to problem nature, size, complexity, applicability and objective/ constraint types.
- The method requires knowledge of orthogonal arrays (and their fractional arrays) and search graph techniques.

Table 5: Description of Different Models Tested. ♣ Standard OAs. ♠ Non-standard OAs.

#-o-V: # of Variables, #-o-C: # of Constraints, N-o-C: Nature of Constraints, Ineq = Inequality, Eq = Equality Constraints, Pr-N: Problem Nature

	Original Model						B. & B. Algorithm	Reduced Model	Size of Array Used		
Problem	#-o-V	#-o-C	N-o-C		Solution	Pr- N	# of Continuous Variables	# of Sub-models	Before relaxation	After relaxation	% of Size reduction
SSWFC_1 [17]	11	17	Ineq		Best	MIP	2	L4OA	L2048♠	L4 OA♣	81.8%
MIMCK_1[5]	30	6			Best	MIP	1	L2OA	L1,073 E+09	L2 OA	96.6%
Max_MCEMS_1 [17]	30	21			Best	0-1	16	L65,536	L1,073 E +09	L65,536 OA ♠	46.67%
NASA_1[17]	14	13	Eq		Good Enough	0-1	6	No relaxations	L16,384♠	L64 OA♣	57.14%
AACS_1[17]	15	12			Best	0-1	0	-	L32,768	-	-
CDOT_1[17]	18	9	Eq/ Ineq		Best	0-1	6	L64 OA – L32 OA - L16OA	L262,144♠	L64 OA♣	66.67%
Min_MCEMS_1 [17]	10	20	Ineq		Best	0-1	0	-	L1024	-	-
Cam-Assignment_1 [17]	44	20	Eq		Best	0-1	0	-	L∞ ♠	-	-
Mitch_1[13]	2	2	Ineq		Best	General Integer	2	L4OA	L4OA	L4OA♣	0.0%
Waste_1[17]	30	24	Eq/ Ineq		Best	MIP	6	L64 OA – L32 OA L16OA L8OA	L∞	L64OA♣	20%

Table 6: Illustrative description of different arrays used & their relative cost.

Nature of Array	**Knowledge of Search Graph**	**2-Levels Arrays**	**Full Array**		**Fractional Array**
		Array Size	# of variables	Type of Array	# of variables (Fractional Array)
Non-standard	√	L2OA	1		
Standard	√	L4OA	2	Low Expense	3
	√	L8OA	3		4
	√	L16OA	4	↑	5,6 (1/2, ¼)
	√	L32OA	5		6,7,8,9 (1/2, ¼, 1/8, 1/16)
	√	L64OA	6		7,8,9,10,11 (1/2, ¼, 1/8, 1/16, 1/32)
Non-standard	×	L128OA	7	Full Factorial	8, 9, 10, 11, 12 (1/2, ¼, 1/8, 1/16, 1/32)
	×	L256OA	8	↓	9, 10, 11, 12, 13, 14 (1/2, ¼, 1/8, 1/16, 1/32, 1/64)
	×	L512OA	9		11, 12, 13, 14, 15
	×	L1024OA	10	High Expense	12, 13, 14, 15 (1/2, ¼, 1/8, 1/16)
	×	L2048OA	11		

√: available
×: not available

Quality of Solutions Obtained	
Full Size Arrays	**Fractional Size Arrays**
1. High quality solution.	1. Low to moderate quality solution
2. Unique (best solution).	2. Non-unique solution.
3. May be expensive for larger size problems.	3. Less expensive for larger size problems

REFERENCES

[1] Bin, Yu and Baozong, Yuan "A More Efficient Branch and Bound Algorithm for Feature Selection", **Pattern Recognition**, Vol. 26, n6, 1993, pp. 883-889.

[2] Boris Goldengorin, Diptesh Ghosh, Gerard Sier Ksma, "Branch and Peg Algorithms for The Simple Plant Location Problem", **Computers and Operations Research**, Vol. 31, 2004, pp. 241-255.

[3] Chao-Y Gau, Linus E. Schrage "Implementation and Testing of a Branch and Bound Based Method for Deterministic Global Optimization: Operations Research Applications", **Frontiers In Global Optimization, C.A. Floudas and P.M. Pardalos, Editors**, 2003, Kluwer Academic Publishers, pp. 1-20.

[4] Ghaith Rabadi , Mansoreeh Mollaghasemi, Georgios C. Anognostoploulous "A Branch and Bound Algorithm for the Early/ Tardy Machine Scheduling Problem with a Common Due Date and Sequence Dependent Setup Time", J. of Computers & Operation Research, 2003, pp. 1-25.

[5] George Kozanidis and Emanuel Melachrionoudis " A Branch & Bound Algorithm for the 0-1 Mixed Integer Knapsack Problem with Linear Multiple Choice Constraints", Computers & Operations Research, Vol. 31, 2004, pp. 695-711.

[6] Gadallah, M.H., Ameer Al-Salem, Khalifa Al-Khalifa, Tamer A. Mohamed, "A Modified Branch & Bound Algorithm via Orthogonal Array Based Enumeration Techniques – Background Method & Preliminary Results", 7th World Congress on Structural and Multidisciplinary Optimization, Coex Seoul, Korea, 2007.

[7] James, R.J.W and Storer, R.H. "Techniques for Solving Subset Sum Problems Within a given Tolerance", **International Transactions in Operational Research**, Vol. 12, 4, 2005, pp. 437-453.

[8] Kent Andersen, Gerard Cornuejols, Yanjun, Li "Reduce and Split Cuts: Improving the Performance of Mixed Integer Gomory Cuts", **Management Science**, Vol. 51 no. 11, 2005, pp. 1720-1732.

[9] Klincewicz, J.G. "Enumeration and Search Procedure for a Hub Location Problem with Economies of Scale", Annals of Operations Research, Vol. 110, 2002, pp. 107-122.

[10] Klincewicz, J.G. "Enumeration and Search Procedure for a Hub Location Problem with Economies of Scale", **Annals of Operations Research**, Vol. 110, 2002, pp. 107-122.

[11] Lieshout, P.M.D and Volgenant, A. "A Branch and Bound Algorithm for the Singly Constrained Assignment Problem", **European Journal of Operational Research**, Vol. 176, n1, 2007, pp. 151-164.

[12] Liming Cai, Juedes D. , Kanji, I. "The In-approximability of Non-Hard Optimization Problems", **Theoretical Computer Science**, Vol. 289 # 1, 2002, pp. 553-571.

[13] Lieshout, P.M.D and Volgenant, A. "A Branch and Bound Algorithm for the Singly Constrained Assignment Problem", European Journal of Operational Research, Vol. 176, n1, 2007, pp. 151-164.

[14] Mitchell, John E. "Branch and Cut Algorithms for Combinatorial Optimization Problems" – Handbook of Applied Optimization, Oxford University Press, 2000.

[15] Mitchell, John E. "Restarting After Branching in the SDP Approach to Max-Cut and Similar Combinatorial Optimization Problems", J. of Combinatorial Optimization, Vol. 5(2), pp. 151-166.

[16] Montemanni, R., L.M. Gambardella "A Branch and Bound Algorithm for the Robust Spanning with Interval Data", European Journal of Operational Research, 2003, pp. xxx-xxx.

[17] Mavrotas, G. and Diakoulaki, D. "Branch and Bound Algorithm for Mixed Zero One Multiple Objective Linear Programming" , European Journal of Operational Research, Vol. 107, n3, 1998, pp. 530-541.

[18] Mitchell, John E. "Restarting After Branching in the SDP Approach to Max-Cut and Similar Combinatorial Optimization Problems", **J. of Combinatorial Optimization**, Vol. 5(2), pp. 151-166.

[19] Mavrotas, G. and Diakoulaki, D. "Branch and Bound Algorithm for Mixed Zero One Multiple Objective Linear Programming" , **European Journal of Operational Research**, Vol. 107, n3, 1998, pp. 530-541.

[20] Pardalos, P.M. and Rodgers, G.P. "Computational Aspects of a Branch and Bound Algorithm for Quadratic Zero One Programming", **Computing**, Vol. 45, n2, 1990, pp. 131-144.

[21] Prticia C., Pop, W. Kern, G. Still, "A New Relaxation Method for the Generalized Minimum Spanning Tree Problem", **European J. of Operational Research**, Vol. 170, 2004, pp. 900-908.

[22] Ronald L. Rardin, **Optimization In Operations Research**, Prentice Hall, Upper Saddle River, New Jersey, 2000.

[23] Ross, Phillip, J. **Taguchi Techniques for Quality Engineering: Loss Function, Orthogonal Experiments, Parameter and Tolerance Design**, Mc Graw Hill Co., New York, 1988.

[24] Roger Z.R.-Mercado, Jonathan F. Bard "A Branch and Bound Algorithm for Permutation Flow Shops with Sequence Dependent Setups", **IIE Transactions**, 1999, Vol. 31, pp. 721-731.

[25] Shi, Chenggen, Lu, Jie, Zhang, Guangquan, Zhou, Hong "An Extended Branch and Bound Algorithm for Linear Bi-level Programming", **Applied Mathematics and Computation**, Vol. 80, n2, 2006, p. 529-537.

[26] Sherali, H.D., Krishnamurty, R.S., Al-Khayyal, F.A. "Enumeration Approach for Linear Complementarity Problems based on a Reformulation Linearization Technique", **J. Optimization Theory and Applications**, Vol. 99, 1998, pp. 481-507.

[27] Simon, de Givry, Laurent Jeannin, "A Unified Framework for partial and Hybrid Search Methods in Constraint Programming", **Computers & Operations Research**, Vol. 33, 2006, pp. 2805-2833.

[28] Toth, P., "Optimization Engineering Techniques for the Exact Solution of NP Hard Combinatorial Optimization Problems", **European Journal of Operational Research**, Vol. 125, 2000, pp. 222-238.

[29] Umit Akinc "Approximate and Exact Algorithms for the Fixed Charge Knapsack Problem", **European Journal of Operational Research**, Vol. 170, 2006, pp. 363-375.

[30] Van Dat Cung, Hi Fi Mi, Le Cun B., "Constrained Two Dimensional Cutting Stock Problems: A Best First Branch and Bound Algorithm", **International Transactions in Operations Research**, Vol. 7, 3, 2000, pp. 185-210.