An Approach to Operations Research Techniques and constraint Programming

Akash Pandey[1], Subhash Kumar Sharma[2], Umesh Kumar Gupta[3]
[1]*Assistant Professor Department of Mathematics, MGPG College, Gorakhpur*
[2]*Assistant Professor, Department of Electronics, MGPG College, Gorakhpur*
[3]*Associate Professor, Department of Mathematics, MGPG College, Gorakhpur*

***Abstract* - The mathematical points of interest and the explicit techniques used to manufacture and investigate these models can be very modern and are tended to. Here we present an overview of the integration of constraint programming (CP) and operations research (OR) to solve combinatorial optimization problems. We interpret CP and OR as relying on a common primal-dual solution approach that provides the basis for integration using four main strategies. The first strategy tightly interweaves propagation from CP and relaxation from OR in a single solver. The second applies OR techniques to domain filtering in CP. The third decomposes the problem into a portion solved by CP and a portion solved by OR, using CP-based column generation or logic-based Benders decomposition. The fourth uses relaxed decision diagrams developed for CP propagation to help solve dynamic programming models in OR. The paper cites a significant fraction of the literature on CP/OR integration and concludes with future perspectives**

***Index Terms* - Constraint Programming, Operations Research, Operation Research Techniques.**

1.INTRODUCTION

Operations Research (O.R.) is a discipline that provides scientific methods for the purpose of solving real life problems that helps us in determining the best utilization of limited resources. Here we study about optimization techniques. In everyday life, we observe many situations of optimization around us. For example, suppose we want to maximize the profit or minimize the cost then maximization of the profit or minimization of cost is the optimization of profit/cost. In O.R., we obtain the optimal solution for decision making problems with the help of optimization techniques. This chapter contains origin, definitions and scope of Operations Research. In this unit, we also discuss the concept of convex sets.

1.1.1. Objectives.

The objective of these contents is to get familiar reader with Operations Research. After studying this unit, reader should be able to define/describe the following concepts like:

- What is Operations Research?
- Origin of Operations Research.
- Scope of operation research.
- Convex Set.

1.2. ORIGIN AND DEFINITIONS OF OPERATIONS RESEARCH

Origin:

Operations Research came into existence and gained prominence during the World War II in Britain with the establishment of team of scientists to study the strategic and tactical problems of various military operations. Scientists of different disciplines were part of this team, their research on military operation soon find applications in other fields also. Now, it was started applying in the fields of industry, trade, agriculture, planning and various other fields of economy and named as 'Operations Research'. Hence the scientific methods and techniques of Operations Research became equally useful for the planners, economists, administrators, irrigation or agricultural experts and statisticians etc. The use of Operations Research has not limited to the Britain only. Many countries of the world had started using O.R. India was one of the few first countries who started using O. R. Regional Research Laboratory located at Hyderabad was the first Operations Research unit established in India during 1949. With the opening of this unit Operations Research in India came into existence. At the same time one more unit was set up in Defence Science Laboratory. In 1955, Operations Research Society of India was formed. Today, O.R. became a professional discipline and studied as a popular subject in Management institutes and school of Mathematics.

Definitions:
Operations Research can be defined simply as combination of two words operation and research where operation means some action applied in any area of interest and research imply some organized process of getting and analysing information about the problem environment. However, many scientists or experts has been defined O.R. in various ways but the opinions about the definitions of it have been changed according to the growth of the subject. So before defining O.R. it is important to see few definitions of it.

1. O.R. is a scientific method of providing executive departments with a quantitative basis for decisions regarding the operations under their control. -Morse and Kimbal (1946)
2. O.R. is a scientific method of providing executive with an analytical and objective basis for decisions. -P.M.S. Blackett (1948)
3. O.R. is the application of scientific methods, techniques and tools to problems involving the operations of system so as to provide these in control of the operations with optimum solutions to the problem. -Churchman, Acoff, Arnoff (1957)
4. O.R. is a management activity pursued in two complementary ways one-half by the free and bold exercise of commonsense untrammeled by any routine, and other half by the application of a repertoire of well-established pre created methods and techniques. -Jagjit Singh (1968)

On the basis of all above opinions, Operations Research can be defined in more general and comprehensive way as: “Operation research is a branch of science which is concerned with the application of scientific methods and techniques to decision making problems and with establishing the optimal solutions"

1.3. SCOPE OF OPERATIONS RESEARCH

Scope of O.R. is very wide in today’s world as it provides better solution to various decision-making problems with great speed and efficiency. Areas where methods/models developed in Operations Research can be applied are given here under:

1. In Agriculture: With the explosion of population and consequent shortage of food, every country is facing the problem of optimum allocation of land to various crops in accordance with the climatic conditions, optimum distribution of water from different resources. Problems of agriculture production under various restrictions can be solved by applications of Operations Research techniques.
2. In Defence Operations: Since Second World War operation research have been used for Defence operations with the aim of obtaining maximum gains with minimum efforts.
3. In Finance: In these modern times, government of every country or every organisation wants to introduce such type of planning/policies regarding their finance and accounting which optimize capital investment, determine optimal replacement strategies, apply cash flow analysis for long range capital investments, formulate credit policies, credit risk. Techniques developed in O.R. can be applied for attaining above said things.
4.In Marketing: A Marketing Administrator has to face many problems like production selection, formulation of competitive strategies, distribution strategies, selection of advertising media with respect to cost and time, finding the optimal number of salesmen, finding optimum time to launch a product. All such problems can be overcome using Operations Research Techniques.
5.In Personnel Management: Every organization wants to make selection of personnel on minimum salary. It needs to find the best combination of workers in different categories with respect to costs, skills, age and nature of jobs. It also needs to frame recruitment policies, assign jobs to machines or workers.
6.In LIC: Operations Research Techniques can be fruitfully applied in LIC offices as it enables the policy makers to decide the premium rates for various modes of policies.
7.In Research and Development: In determination of the areas of concentration of research and development. It also helps in project selection.

O.R. helps in solving many other problems faced by public as well as private sectors such as the ones in economic and social planning, management of natural resources, energy, housing pollution control, waiting lines and administrative problems, insurance policies and many more.

1.4. ADVANTAGES OF OPERATIONS RESEARCH

Following are certain advantages of Operations Research (OR):

• Operations Research helps decision –maker to take better and quicker decisions. It helps decision –maker to evaluate the risk and results of all the alternative decisions. So, it improves the quality of decisions and makes the decisions more effective.

• Operation Research helps, in preparing future managers as it provides in-depth knowledge about a particular action.

• Operations Research develop models, which provides logical and systematic approach for understanding, Solving and controlling a problem.

• Operations research reduces the chances of failure as it provides many alternatives for one problem, which helps the management to choose the best decision. Even managers can evaluate the risks associated with each solution and can decide whether they want to go with the solution or not.

• It helps users in optimum use of resources. For example, linear programming techniques in Operations Research suggest most effective methods and efficient ways of optimality.

• It helps in finding the limitations and scope of an activity.

• Using this information, he can measure the performance of employees and can compare it with the standard performance. It modifies mathematical solutions before these are applied. Managers may accept or modify the mathematical solutions obtained using Operations Research techniques.

• It helps suggest alternative solutions for the same optimum profit if the management wants so.

1.5 LIMITATIONS OF OPERATIONS RESEARCH

• Formulation of mathematical models may take into account all possible factors for defining a reallife problem and hence is difficult. As a result, the help of computers is required for the large number of cumbersome computations for such problems. This discourages small companies and other organisations from using O.R. techniques.

• Unquantifiable factors: Some problems may involve a large number of intangible factors such as human emotions, human relationship, etc. which cannot be quantified. Hence, the best solution cannot be determined for such problems because such factors have to be excluded.

•Dependence on experts: A specialist, who may be a mathematician or a statistician, is needed to understand the formulation of models, find solutions and recommend their implementation. Managers, who deal with such problems, may not have such specialisation. Managers, who deal with such problems, may not have such specialisation and hence the results may not be optimal.

•Model is abstraction of real-life situations and not the reality.

•Assumptions need to be made about the nature and importance of some factors in order to construct an Operation Research model.

•A reasonably good solution without the use of Operation Research may be preferred by the management as compared to a slightly better solution provided by using Operation Research since it is very expensive in terms of time and money. In the next chapter onwards, we shall introduce various O.R. techniques for obtaining optimal and feasible solutions. Before studying these techniques, you must familiar with some important basic concepts like convex sets and basic feasible solutions. Now, we will discuss these concepts.

1.6. CONVEX SET

A region or a set K is convex if and only if for any two points on the set K, the line segment connecting these points lies entirely in K. Mathematically, $(x_1\ y_1) \in K$.

$(\lambda x_1 + (1-\lambda)\ x_2, \lambda y_1 + (1-\lambda) y_2) \in K \quad ,0 \leq \lambda \leq 1$

Where $(\lambda x_1 + (1 - \lambda)_2, \lambda y_1 + (1 - \lambda)\ y_2)$ gives all the points which lie on the line segment joining $(x1, y1)$ and $(x2, y2)$.

Example of a Convex Set

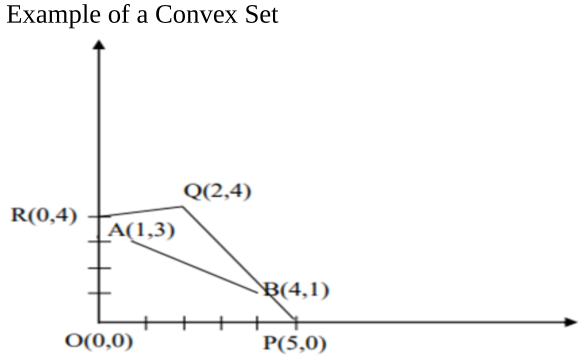

Fig (1.1)

Consider the region enclosed by OPQR. Let us denote it by K. It is convex as the line segment joining any two points in this region lies wholly within it. As an example, let us take two points A (1, 3) and B (4, 1).

Then all points on the line segment joining A and B are given by
$((1) + (1-\lambda)4, \lambda(3) + (1-\lambda)(1)) = (\lambda + 4 - 4\lambda, 3\lambda + 1 - \lambda) = (4 - 3\lambda, 1 + 2\lambda), 0 \leq \lambda \leq 1$.
Here $\lambda = 0$ gives the point Q (4, 1) and $\lambda = 1$ gives the point P (1, 3). Other points on the line segment AB are given by = (4 - 3 λ, 1 + 2 λ), where 0 < 1
For example, let us take λ=0.1, 0.3, 0.5, 0.7, 0.9; then the corresponding points (after substituting the values of λ = 0.1, 0.3, 0.5, 0.7, 0.9) are; (4 – 3(0.1), 1 + 2(0.1)), (4-3× 0.3, 1 + 2 × 0.3), (4- 3× 0.5, 1 + 2 × 0.5), (4-3× 0.7, 1 + 2 × 0.7), (4-3× 0.9, 1 + 2 × 0.9) i.e., (3.7,1.2), (3.1 ,1.6), (2.5,2), (1.9, 2.4), (1.3, 2.8) All these points clearly lie on the line and also in the region K. Similarly, all other points on the line segment AB also lie inside the region K. Hence, the line segment AB lies in K. Therefore, K is convex in this example.
Example of Non-Convex Set
1.6.1. Example.

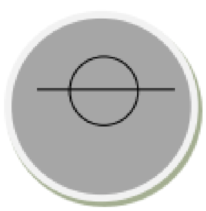

Fig (1.2)

Consider the shaded region in Fig. 1.2, clearly the line segment joining two points do not lie wholly in the region and hence this is an example of non-convex set.
1.6.2. Example. Show that the set T = {(x, y): $x^2 + y^2 \leq 1$} is a convex set.
Solution: Let us take any two points A (x_1, y_1) and B (x_2, y_2) in Fig. 1.3 such that:
$x_1^2 + y_1^2 \leq 1$, And $x_2^2 + y_2^2 \leq 1$.

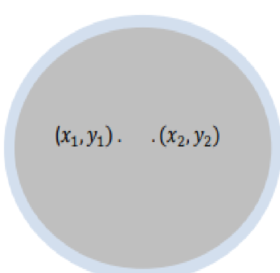

Fig (1.3)

Now, the line segment joining A and B is the set
$\{\lambda x_1 + (1-\lambda)_2, \lambda y_1 + (1-\lambda)y_2 : 0 \leq \lambda \leq 1\}$.
Let $u_1 = \lambda x_1 + (1-\lambda)_2$, $u_2 = \lambda y_1 + (1-\lambda)y_2$

Therefore, all points on the line segment AB are given by (u_1, u_2). Now, the line segment AB lies wholly in T if
$u_1^2 + u_2^2 \leq 1$
Since
$u_1^2 + u_2^2 = [\lambda x_1 + (1-\lambda)_2]^2 + [\lambda y_1 + (1-\lambda)y_2]^2$
$= \lambda^2 x_1^2 + (1-\lambda)^2 x_2^2 + 2\lambda(1-\lambda)x_1x_2 + \lambda^2 y_1^2 + (1-\lambda)^2 y_2^2 + 2\lambda(1-\lambda)y_1y_2$
$= \lambda^2[x_1^2 + y_1^2] + (1-\lambda)^2[x_2^2 + y_2^2] + 2\lambda(1-\lambda)[x_1x_2 + y_1y_2]$
We have
$u_1^2 + u_2^2 \leq \lambda^2 + (1-\lambda)^2 + 2\lambda(1-\lambda)[x_1x_2 + y_1y_2]$ (1)
Now consider $(x_1x_2 + y_1y_2)^2 = x_1^2x_2^2 + y_1^2y_2^2 + 2x_1x_2y_1y_2$
$= x_1^2x_2^2 + y_1^2y_2^2 + x_1^2y_2^2 + x_2^2y_1^2 - x_1^2y_2^2 - x_2^2y_1^2 + 2x_1x_2y_1y_2$
$= (x_1^2 + y_1^2)(x_2^2 + y_2^2) - x_1y_2(x_1y_2 - x_2y_1) - x_2y_1(x_2y_1 - x_1y_2)$
$= (x_1^2 + y_1^2)(x_2^2 + y_2^2) - (x_2y_1 - x_1y_2)^2$
$\leq (x_2y_1 - x_1y_2) \leq 1$
$\Rightarrow (x_1x_2 + y_1y_2) \leq 1$
∴ From (1) and (2), we have
$u_1^2 + u_2^2 \leq \lambda^2 + (1-\lambda)^2 + 2\lambda(1-\lambda)$
Or $u_1^2 + u_2^2 \leq [\lambda + (1-\lambda)]^2$
$\Rightarrow u_1^2 + u_2^2 \leq 1$
∴ T is convex set.
1.6.3. Extreme Points of a convex set
A point (x, y) in a convex set K is called an extreme point if it is not possible to locate two distinct points in or on K so that the line joining them will include (x, y). Mathematically, a point (x, y) is an extreme point of a convex set if it cannot be expressed as a convex combination of any two points (x_1, y_1) and (x_2, y_2) [for $(x_1, y_1) \neq (x_2, y_2)$] in the set such that $x = \lambda x_1 + (1-\lambda)_2$ and $y = \lambda y_1 + (1-\lambda)y_2$, 0< 1
Remark:
i) The vertices of the polygons, which are convex sets, are the extreme points.
ii) Every point on the circumference of the region containing the portion in and on the circle is an extreme point.

1.6.4. IMPORTANT DEFINITIONS
Solution
A set of values of the decision variables which satisfy the constraints of the given LPP is said to be a solution of that LPP.

Feasible Solution

A solution in which values of decision variables satisfy all the constraints and non-negativity conditions of an LPP simultaneously is known as feasible solution.

Infeasible Solution

A solution in which values of decision variables do not satisfy all the constraints and non-negativity conditions of an LPP simultaneously is known as infeasible solution.

Basic solution

Suppose there are m equations representing constraints (limited available resources) containing m + n variables in an allocation problem. The solution obtained by setting any n variables equal to zero and solving for the remaining m variables and the remaining n variables are non – basic variables. The maximum number of possible basic solutions is given by the formula $C_n{}^{m+n}$

For example, if there are 2 equations in 3 variables, then the maximum number of possible basic solutions is

$$C_2^3 = \frac{3!}{2!\,(3-2)!} = 3.$$

Basic Feasible Solution

A basic solution for which all the basic variables are non – negative is called the basic feasible solution. Further basic feasible solution are of two types:

Degenerate Solution

A basic feasible solution is known as degenerate if value of at least one basic variable is zero. Non-

Degenerate Solution

A basic feasible solution is known as non- degenerate if value of all basic variables are non-zero and positive.

Optimum Basic Feasible Solution

A basic feasible solution which optimizes i.e. maximise or minimise the objective function value of the given LPP is called optimum basic feasible solution.

Unbounded Solution

A solution in which value of the objective function of the given LPP increase/decrease indefinitely is called an unbounded solution.

1.7 Constraint Programming Concepts:

Constraint programming (CP) and operations research (OR) have the same overall goal. They strive to capture a real-world situation in a mathematical model and solve it efficiently. Both fields use constraints to build the model, often in conjunction with an objective function to evaluate solutions. It is therefore only natural that the two fields join forces to solve problems Attempting to unify CP and OR might be unwise if they relied on entirely different solution methods. However, their methods are not only related, but complementary, due to the contrasting intellectual origins of the two fields. This has allowed integrated methods to outperform those that rely solely on CP or OR techniques in a wide variety of problem areas, sometimes by orders of magnitude. Furthermore, the potential benefits of integration are, arguably, only beginning to be reaped, which suggests that CP/OR integration will continue to be an active research area. Both CP and OR use what the OR community might call a primal-dual approach, which combines search with some kind of inference. Search solves the primal problem of finding a feasible solution (one that satisfies the constraints), while inference solves the dual problem of proving that a solution is optimal, or that no solution is feasible. Search frequently takes the form of a branching mechanism, at least in the context of exact methods. The two fields diverge when it comes to inference. In OR, it typically appears as problem relaxation, strengthened by such inferred constraints as cutting planes. In CP, inference appears as constraint propagation and domain filtering. Both relaxation and propagation can help find feasible solutions as well. Operations research is strongly influenced by its historical roots in linear programming (LP), which formulates problems using inequality constraints. Much of the field today is based on inequality-constrained mathematical programming models, including those of nonlinear programming (NLP), mixed integer/linear programming (MILP), and mixed integer/nonlinear programming. A model is almost always relaxed by reducing it to a simpler inequality constrained model, such as an LP or a convex NLP model, which can be solved with highly developed methods that exploit its special structure.

Relaxation is essential because it allows the solver to infer a bound on the optimal value, which reduces branching. The relaxation is often strengthened by valid inequalities that are inferred from the constraint set. Operations research is, of course, broader than mathematical programming, as it encompasses dynamic programming, queuing theory, simulation, and other areas. Although we focus primarily on mathematical programming, we will see that dynamic programming, as well as network and matching theory, also play a significant role in CP. Constraint programming has roots in logic programming, where a model has both a declarative and a procedural interpretation. A model is declarative because its statements can be read as logical propositions that describe the problem, and it is procedural because the statements can be processed as instructions for how to find a solution. Something similar to this dual interpretation survives in today's CP. The statements in a model impose constraints that describe the problem, even while they invoke algorithms, such as domain filtering, that lead to a reduction in branching. Due to these contrasting origins, OR and CP process a model differently as they conduct a search. OR solves an inequality-constrained relaxation of the model as a single problem, while CP processes the constraints of the model individually. This allows OR to combine information from the entire model while inferring a bound, but relaxation sacrifices much of the combinatorial complexity of the problem. The CP approach captures much of the combinatorial complexity of individual constraints while inferring reduced domains, but it must resort to the propagation of domains from one constraint to the next to obtain a global view. OR partially compensates for the weakness of its relaxations by strengthening them with valid constraints that capture some of the special structure of groups of constraints. CP partially compensates for the weakness of constraint propagation by defining high-level global constraints that represent a group of simpler constraints. At least four basic strategies for combining the complementary strengths of OR and CP have been developed in the literature. They can be summarized as follows. Combine relaxation from OR with propagation from CP. This can be effective when some constraints "relax well" in the sense that they have a tight inequality relaxation, and others "propagate well." A relaxation is tight when its feasible set is similar to that of the original problem, or at least yields a similar optimal value. Constraints propagate well when their structure allows significant domain reduction when some variables are fixed (or their domains reduced), perhaps by branching. The so-called knapsack constraints of OR, which are linear inequalities with many nonzero coefficients, tend to relax well, because they serve as their own LP relaxation. Certain groups of constraints can also give rise to useful valid inequalities, such as the famous Gomory cuts, which are derived from constraints that are tight in the solution of the LP relaxation, or the valid cuts derived from subtours and "combs" in the solution of a relaxed traveling salesman problem. The classical "binary" constraints of CP, which contain only two variables, generally propagate well, because fixing (or reducing the domain of) one variable tends to have a significant effect on the domain of the other. High-level global constraints may also propagate well, assuming they have been analyzed and implemented in solvers. Examples include disjunctive and cumulative scheduling constraints, which have been deeply analyzed and help explain the success of CP in the scheduling domain. Use OR methods for domain filtering in CP. Network and matching theory, as well as dynamic programming, are widely used to filter domains for a variety of global constraints. Edge-finding methods, originally developed in OR, are indispensable for domain filtering in disjunctive and cumulative scheduling problems. In addition, since achieving domain consistency for a global constraint is frequently an NP-hard problem, it can be helpful to use a more tractable OR-based relaxation of the constraint as a basis for filtering. Decompose the problem into parts that suitable for OR and CP, respectively. This can be accomplished with two decomposition methods originally developed in the OR literature: column generation and Benders decomposition. Column generation accommodates CP by using it to generate columns in the pricing sub problem. Benders methods can accommodate CP if they are generalized to "logic-based" Benders decomposition, which allows the sub problem to be solved by CP.

Apply constraint propagation to dynamic programming models. If a problem can be given a recursive model as in dynamic programming, the statetransition graph for the model can be treated as a decision diagram, and arcs can be deleted from the diagram much as values from a domain in a conventional domain store. Normally a relaxed decision diagram is used rather than an exact one, which tends to grow exponentially. The relaxed diagram not only allows propagation that is stronger than propagation through domains, but it provides a valid bound on the optimal value that allows one to solve a dynamic programming model by branch-and-bound methods when no inequality-constrained relaxation is available.

1.8 Combining Propagation and Relaxation :

One general strategy for integrating CP and OR is to combine constraint propagation with relaxation. The two techniques are mutually reinforcing, because propagation can tighten bounds on the variables in an LP relaxation while a relaxation can prove optimality or infeasibility of a problem obtained during CP-based search. An early application of this strategy (1995) used an LP relaxation to prove the optimality of a solution obtained by CP in five minutes for a boatparty scheduling problem that MILP could not solve in five hours [1].

Relaxation of Global Constraints: Many constraint programming languages allow the definition of global constraints that represent NP-complete problems. For example, one of the earliest constraint languages, Alice [2], included the 'circuit' constraint to state that a set of variables represent a Hamiltonian circuit in a graph. Since establishing domain consistency for such constraint would be NP-hard, it is natural to design propagation algorithms based on a relaxation of the constraint. In particular relaxations stemming from OR, such as linear programming and Lagrangian relaxations, have been used for this purpose. One of the first systematic applications of linear relaxations in global constraint propagation was developed in a series of papers by Focacci, Lodi, and Milano [3, 4, 5, 6]. Using the traveling salesman problem with time windows as illustrative application, they develop optimization-oriented global constraints that 1) use the linear programming bound to tighten the domain of the variable representing the objective, and 2) apply reduced-cost based variable fixing to filter sub-optimal domain values. Reduced costs can also be applied to guide and decompose the CP search [7]. In these applications, the global constraint provides an interface for the finite-domain variables in the CP model to the continuous variables in the associated linear programming model.

1.9 Linear Relaxations from CP Models:

In the previous section, linear or Lagrangian relaxations are inferred from individual (global) constraints, which represent a specific combinatorial structure. This approach can be generalized to arbitrary subsets of constraints, or even the entire problem. That is, for a given CP (sub)problem, we can create a linear programming model that serves as a relaxation to the problem. Such linear model can then be maintained during search and applied for improved optimization bounds, reduced cost based variable fixing, or guiding the search. The first systematic approaches to automatically reformulate CP models into linear programming models were proposed for this purpose by Rodosek et al. [8] and Refalo [9]. The approach was further developed and implemented in the eplex library of the constraint logic programming system Eclipse in [10]. Belov et al. [11] present a related work that automatically translates CP models in MiniZinc to equivalent linear MIP models, to be solved by MIP solvers. However, such generic transformations may lead to poor LP relaxations, especially when many 'big-M' constraints are needed. Stronger linear models may be derived by taking into account the semantic information in CP models. In particular, Laborie and Rogerie present an automatically generated linear relaxation for advanced scheduling models, as used in IBM ILOG CP Optimizer. This LP relaxation can be particularly helpful for complex objective functions [12]. Naturally, the strongest possible LP relaxations can be derived for specific applications. In addition to a tailored linear model, this also allows the addition of problem-specific cuts to strengthen the relaxation. Example applications in which dedicated LP relaxations are embedded as a global constraint in CP models include multi-agent scheduling [13], integrated employee timetabling and job-shop scheduling [14], and time-dependent sequencing problems [15].

OR-Based Filtering Methods:

OR methods have made major contributions to domain filtering for global constraints in CP. Outstanding examples include the all-different constraint, the generalized cardinality constraint, disjunctive and cumulative scheduling constraints, the sequence constraint, and the stretch constraint. The all-different constraint first appeared in 1978 [16]. Filtering algorithms that achieve domain consistency for all-different were derived in the early 1990s [17] using results from matching theory in the OR literature [18, 19, 20], which is in turn based on classical network flow theory. The OR literature also provided the basis for achieving bounds consistency [21], namely a result for convex graphs [22]. The generalized cardinality constraint is filtered using a network flow model [23], and bounds consistency achieved using a flow-based algorithm that again exploits convexity of the graph.In the 2000s the network-flow based propagation was extended to costbased global constraints, by representing them with minimum-cost network flows. This was first done to establish domain consistency for weighted cardinality constraints. Minimum-cost network flows have also been applied to establish domain consistency on soft global constraints for which one aims to minimize the violation, as was first done for the soft all-different constraint. This approach was generalized and applied to soft cardinality and soft regular constraints. An overview of soft global constraints can be found in. Other global constraints that use minimum-cost network flows include the soft sequence constraint, the soft all-different constraint with preferences, the soft cardinality and soft regular constraints with Preferences, soft global constraints for weighted CSPs, and soft open global constraints. Disjunctive and cumulative scheduling represent one of the key successes of OR/CP collaboration. It began with the edge-finding algorithms of Carlier and Pinson, published in the OR literature. These algorithms reduce the time windows within which tasks must execute, based on the fact that they cannot overlap, and thereby accelerate the search for a feasible schedule. The technology then passed over to the CP community, which further developed edge-finding methods for disjunctive scheduling and extended them to allow incremental updates and setup times. These were followed by not-first/not-last rules, which achieve some bound tightening missed by edge finding. In the meantime, the cumulative scheduling constraint was introduced, which along with its variations, became a major component of CP's powerful scheduling technology. A number of edge-finding algorithms for the constraint appeared, along with "extended" edge finding, not-first/not-last rules and energetic reasoning. Much of this work is described. Although these contributions advanced substantially beyond the original edge-finding methods, they owe their intellectual inspiration to ideas that came out of the OR literature. The sequence constraint also illustrates a remarkable linkage of OR and CP. While there are elegant polynomial-time filters for achieving domain consistency that do not rely on OR methods a competitive polytime filtering method is grounded in deep results from integer programming. An integer programming model for the constraint has a coefficient matrix that exhibits the consecutive ones property, which means that the matrix is totally unimodular, and the problem can be solved by LP alone. Furthermore, it is known that such a problem can be given a specially-structured LP formulation, namely a rather unobvious network flow model. This provides the basis for an efficient polytime filtering algorithm.

2.COLUMN GENERATION

Some linear programming models consist of a huge number of variables, as compared to the number of constraints – perhaps the size of the model even exceeds the memory of the computer. It is still possible to solve such LPs efficiently, by recognizing that an optimal solution only needs at most as many non-zero variables as the number of constraints. Namely, we can start with an initial (small) subset of variables that permits a feasible LP solution. After solving the LP, we identify a new variable that may improve the current solution by evaluating its reduced cost. We then add the new variable to the LP model and repeat. If there are no variables with a negative reduced cost (for a minimization problem), the current solution is optimal, by LP theory. This decomposition approach is called column generation, as variables correspond to columns in the matrix representation of LP models. The LP defined by the current set of columns is called the (restricted) master problem. Finding a new variable consists of finding the entries of its column, i.e., the coefficients of the linear constraints in which the variable appears, which is done in the pricing problem. Column generation can also be applied to integer linear programming models, by embedding the procedure in an enumerative earch called branch-and-price. It is one of the most important and widely used OR techniques for large-scale optimization.

2.1Benders Decomposition

Benders decomposition is designed for problems that yield a much simpler problem (the Benders subproblem) when certain variables are fixed. The subproblem is solved to obtain one or more Benders cuts that bound the cost of fixing variables to these or similar values. The Benders cuts are added to a master problem that is solved to find the next set of values for the fixed variables. The process is repeated until the optimal values of the master problem and subproblem converge. Thus the problem decomposes into two parts that communicate through Benders cuts. In the original Benders method, the sub problem must be an LP, and the Benders cut is derived from the LP dual. Hooker and Hooker and Ottosson substantially generalized the classical method to logicbased Benders decomposition, in which the subproblem can in principle be any optimization or constraint satisfaction problem, and the Benders cuts are derived from an inference dual. Logic-based Benders decomposition (LBBD) provides a broad scope for OR/CP collaboration, because the master problem and subproblem can be attacked with different solvers, one from OR and one from CP. In most applications, the subproblem is a CP problem, perhaps a scheduling problem. Its combinatorial nature is no longer a barrier to generating Benders cuts. The master problem can be solved by whatever OR method is convenient, such as MILP or a heuristic method. Logic-based cuts must be developed anew for each problem class, unlike classical Benders cuts, which are always based on the LP dual in the same way. However, this provides an opportunity to exploit the special structure of the problem.

2.3Decision Diagrams and Dynamic Programming:

Decision diagrams have long been used for circuit design and product configuration. More recently, Hadˇzi´c and Hooker adapted decision diagrams to optimization and, with Anderson and Tiedemann ,showed that they can be an effective alternative to the traditional constraint store in CP. Rather than propagate through variable domains, one can propagate through a decision diagram that represents a discrete relaxation of the prob- 14 J. N. Hooker and W.-J. van Hoeve lem. The connection of decision diagrams to operations research is that they are well suited for the solution of optimization problems that have dynamic programming models. Dynamic programming models are normally solved by a recursive process that enumerates the state space at each stage of the recursion. Because the state space typically grows exponentially with the number of state variables, such techniques as state space approximation and approximate dynamic programming are often used to resist the “curse of dimensionality” .Decision diagrams provide the option of solving the problem by a branchand-bound technique, and in particular, one that branches on nodes of a relaxed decision diagram rather than on values of variables .The bounding mechanism is based on relaxation values obtained from relaxed sub-diagrams rooted at branching nodes, much as traditional branch and bound is based on relaxation bounds obtained from LP relaxations at nodes of the branching tree. This can lead to significant speedups relative to state-of-the-art MILP solvers on some problems that have a natural MILP formulation .Its primary potential, however, is in the solution of dynamic programming models that are not readily formulated as MILP problems. Viewing a dynamic programming model in terms of decision diagrams can occasionally lead to radical simplification of the problem. This is accomplished by rearranging costs on the arcs of the decision diagram (which are immediate costs in the dynamic programming model) so that the diagram can be reduced to a much simpler diagram.

3.CONCLUSIONS

In this paper, we have introduced Operations Research, its scope, advantages and limitations. We have observed that Operations Research is a very powerful method of getting the best out of limited resources. It finds applications in almost every field. Here, we explain concept of convex sets which is another important concept. We study feasible solution, basic solution, and basic feasible solution of a system of equations less in number than the number of decision variables. Such solutions are required to be obtained for finding out optimal solution of the given LPP The integration of CP and OR has proceeded over a period of nearly three decades, first rather slowly, but at a gradually quickening pace. It has brought improved solution methods—sometimes radically improved—to a wide variety of problems, as well as advances in modelling. The perspective afforded by one field has lent new insight into the other, which in turn leads to still more effective methods. Despite this considerable progress, there remains great potential for further integration, with the concomitant improvement in both modelling and solution methods. Any attempt to predict the direction of research is a fool's errand, but we can point out some current research activity that shows promise for further progress, as well as some possible areas for future research. One active area of current research is the development of advanced modelling systems that invoke both CP and OR solvers.

REFERENCES

[1] Smith, B.M., Brailsford, S.C., Hubbard, P.M., Williams, H.P.: The progressive party problem: Integer linear programming and constraint programming compared. In: U. Montanari, F. Rossi (eds.) Principles and Practice of Constraint Programming (CP 1995), Lecture Notes in Computer Science, vol. 976, pp. 36–52. Springer (1995).

[2] Lauri`ere, J.L.: A language and a program for stating and solving combinatorial problems. Artificial Intelligence 10, 29–127 (1978).

[3] Focacci, F., Lodi, A., Milano, M.: Cost-based domain filtering. In: J. Jaffar (ed.) Principles and Practice of Constraint Programming (CP 1999), Lecture Notes in Computer Science, vol. 1713, pp. 189–203. Springer (1999).

[4] Focacci, F., Lodi, A., Milano, M.: A Hybrid Exact Algorithm for the TSPTW. INFORMS Journal on Computing 14, 403–417 (2002).

[5] Focacci, F., Lodi, A., Milano, M.: Embedding Relaxations in Global Constraints for Solving TSP and TSPTW. Annals of Mathematics and Artificial Intelligence 34,291– 311 (2002).

[6] Focacci, F., Lodi, A., Milano, M.: Optimization-Oriented Global Constraints. Constraints 7, 351–365 (2002).

[7] Milano, M., van Hoeve, W.J.: Reduced cost-based ranking for generating promising subproblems. In: P. Van Hentenryck (ed.) Principles and Practice of Constraint Programming (CP 2002), Lecture Notes in Computer Science, vol. 2470, pp. 1–16. Springer (2002).

[8] Rodoˇsek, R., Wallace, M., Hajian, M.: A new approach to integrating mixed integer programming and constraint logic programming. Annals of Operations Research 86, 63–87 (1999).

[9] Refalo, P.: Linear formulation of constraint programming models and hybrid solvers. In: R. Dechter (ed.) Principles and Practice of Constraint Programming (CP 2000), Lecture Notes in Computer Science, vol. 1894, pp. 369–383. Springer (2000).

[10] Shen, K., Schimpf, J.: Eplex: Harnessing Mathematical Programming Solvers for Constraint Logic Programming. In: Proceedings of CP, Lecture Notes in Computer Science, vol. 3709, pp. 622–636. Springer (2005).

[11] Belov, G., Stuckey, P.J., Tack, G., Wallace, M.: Improved Linearization of Constraint Programming Models. In: Proceedings of CP, Lecture Notes in Computer Science, vol. 9892, pp. 49–65. Springer (2016).

[12] Laborie, P., Rogerie, J.: Temporal linear relaxation in IBM ILOG CP Optimizer. J. Scheduling 19(4), 391–400 (2016).

[13] van Hoeve, W.J., Gomes, C.P., Selman, B., Lombardi, M.: Optimal Multi-Agent Scheduling with Constraint Programming. In: Proceedings of AAAI, pp. 1813–1818. AAAI Press (2007)

[14] Artigues, C., Gendreau, M., Rousseau, L.M., Vergnaud, A.: Solving an integrated employee timetabling and job-shop scheduling problem via hybrid branch-and-bound. Computers & Operations Research 36, 2330–2340 (2009).

[15] Kinable, J., Cire, A.A., van Hoeve, W.J.: Hybrid optimization methods for timedependent sequencing problems. European Journal of Operational Research 259, 887– 897 (2017).

[16] Lauri`ere, J.L.: A language and a program for stating and solving combinatorial problems. Artificial Intelligence 10, 29–127 (1978).

[17] R´egin, J.C.: A filtering algorithm for constraints of difference in CSP. In: National Conference on Artificial Intelligence (AAAI 1994), pp. 362–367. AAAI Press (1994).

[18] Hopcroft, J.E., Karp, R.M.: A n 5/2 algorithm for maximum matchings in bipartite graphs. SIAM Journal on Computing 2, 225–231 (1973).

[19] Lov´asz, L., Plummer, M.: Matching Theory, Annals of discrete Mathematics, vol. 29. North-Holland (1986)

[20] Costa, M.C.: Persistency in maximum cardinality bipartite matchings. Operations Research Letters 15, 143–149 (1994).

[21] Mehlhorn, K., Thiel, S.: Faster algorithms for bound consistency of the sortedness and all different constraint. In: R. Dechter (ed.) Principles and Practice of Constraint Programming (CP 2000), Lecture Notes in Computer Science, vol. 1894, pp. 306–319. Springer (2000).

[22] Glover, F.: Maximum matching in a convex bipartite graph. Naval Research Logistics Quarterly 316, 313–316 (1967).

[23] R´egin, J.C.: Generalized arc consistency for global cardinality constraint. In: National Conference on Artificial Intelligence (AAAI 1996), pp. 209–215. AAAI Press (1996).