Solving VRPTWs with Constraint Programming Based Column Generation

LOUIS-MARTIN ROUSSEAU, MICHEL GENDREAU and GILLES PESANT
{louism,michelg,pesant}@crt.umontreal.ca
Centre de recherche sur les transports, Université de Montréal, CP 6128 Succ Centre-Ville, Montréal, Canada H3C 3J7

FILIPPO FOCACCI ffocacci@ilog.fr
ILOG SA 9, rue de Verdun, BP 85, 94253 Gentilly Cedex, France

Abstract. Constraint programming based column generation is a hybrid optimization framework recently proposed (Junker et al., 1999) that uses constraint programming to solve column generation subproblems. In the past, this framework has been used to solve scheduling problems where the associated graph is naturally acyclic and has done so very efficiently. This paper attempts to solve problems whose graph is cyclic by nature, such as routing problems, by solving the elementary shortest path problem with constraint programming. We also introduce new redundant constraints which can be useful in the general framework. The experimental results are comparable to those of the similar method in the literature (Desrochers, Desrosiers, and Solomon, 1992) but the proposed method yields a much more flexible approach.

Keywords: column generation, constraint programming, hybrid method, vehicle routing with time windows, optimization constraints

Introduction

Vehicle Routing Problems (VRP) are widely present in today’s industries, ranging from distribution problems to fleet management. They account for a significant portion of the operational cost of many companies. The VRP can be described as follows: given a set of customers $C$, a set of vehicles $V$, and a depot $d$, find a set of routes of minimal length, starting and ending at $d$, such that each customer in $C$ is visited by exactly one vehicle in $V$. Each customer having a specific demand, there are usually capacity constraints on the load that can be carried by a vehicle. In addition, there is a maximum amount of time that can be spent on the road. The time window variant of the problem (VRPTW) imposes the additional constraint that each customer $c$ must be visited after time $a_c$ and before time $b_c$. One can wait in case of early arrival, but late arrival is not permitted.

Column generation was introduced by Dantzig and Wolfe (1960) to solve linear programs with decomposable structures, it has been applied to many problems with success and has become a leading optimization technique to solve Crew Scheduling Problems (Desrosiers et al., 1995). In the first application to the field of Vehicle Routing Problems with Time Windows, presented by Desrochers, Desrosiers, and Solomon
(1992), the basic idea was to decompose the problem into sets of customers visited by the same vehicle (routes) and to select the optimal set of routes between all possible ones. Letting \( r \) be a feasible route in the original graph (which contains \( N \) customers); \( R \) be the set of all possible routes \( r \), \( c_r \) be the cost of visiting all the customers in \( r \); \( A = (a_{ir}) \) be a Boolean matrix expressing the presence of a particular customer (denoted by index \( i \in [1..N] \)) in route \( r \); and \( x_r \) a Boolean variable specifying whether the route \( r \) is chosen \((x_r = 1)\) or not \((x_r = 0)\), the Set Partitioning Problem is defined as \((S)\):

\[
\min \sum_{r \in R} c_r x_r \\
\text{s.t.} \sum_{r \in R} a_{ir} x_r = 1 \quad \forall i \in [1..N], \quad x \in \{0, 1\}^N.
\]

This formulation, however, poses some problems. Firstly, since it is impractical to construct and to store the set \( R \) because of its very large size, it is usual to work with a partial set \( R' \) that is enriched iteratively by solving a subproblem. Secondly, the Set Partitioning formulation is difficult to solve when \( R' \) is small and it allows negative dual values which can be problematic for the subproblem (a negative dual means a negative marginal cost to service a node). That is why, in general, the following relaxed Set Covering formulation is used instead as a Master Problem \((M)\):

\[
\min \sum_{r \in R'} c_r x_r \\
\text{s.t.} \sum_{r \in R'} a_{ir} x_r \geq 1 \quad \forall i \in [1..N], \quad x \in [0, 1]^N.
\]

To enrich \( R' \), it is necessary to find new routes which offer a better way to visit the customers they contain, that is, routes which present a negative reduced cost. The reduced cost of a route is calculated by replacing the cost of an arc (the distances between two customers) \( d_{ij} \) by the reduced cost of that arc \( c_{ij} = d_{ij} - \lambda_i \) where \( \lambda_i \) is the dual value associated with customer \( i \). The dual value associated with a customer can be interpreted as the marginal cost of visiting that customer in the current optimal solution (given for \( R' \)). The objective of the subproblem is then the identification of a negative reduced cost path, that is, a path for which the sum of the travelled distance is inferior to the sum of the marginal costs (dual values). Such a path represents a novel and better way to visit the customers it serves.

The optimal solution of \((M)\) has been identified when there exists no more negative reduced cost path. This solution can, however, be fractional, since \((M)\) is a relaxation of \((S)\), and thus does not represent the optimal solution to \((S)\) but rather a lower bound on it. If this is the case, it is necessary to start a branching scheme in order to identify an integer solution.