A GRASP for the Surveillance Patrol Vehicle Routing Problem

Simona Mancini[1]

Dipartimento di Scienze, Progetto e Politiche del Territorio, Politecnico di Torino, Italy
simona.mancini@polito.it

1 Introduction

In this paper a new rich vehicle routing problem is presented, the Surveillance Patrol Vehicle Routing Problem (SPVRP). This problem came out from a real needing of a surveillance company to create fairer routing plans for their patrols.The problem consist into routing a set of patrols in order to visit a set of checkpoints. Each checkpoint may require a different number of visits each one to be to be exploited within a given time window. Furthermore, a minimum time spacing between two consecutive visits should be observed. Each checkpoint is characterized by a service time, which represents the time to be spent at the checkpoint during each visit (this time may vary from 1-2 minutes to 15-20 minutes depending on the service required) and is associated with a difficulty level, basing on its degree of danger. Checkpoints located in residential zones generally have a low dangerousness, due to the low number of crimes committed in the zone, while checkpoints located in slum neighborhood or near the railway station or places with an high affluence of people like stadiums, are potentially much more dangerous and must be controlled only by expert patrols. Each patrol is associated with an expertness level, such as a checkpoint may be assigned to a checkpoint only if it holds the minimum experience level required by the checkpoint. The same checkpoint can be visited by different compatible patrols. Obviously more experted patrols cost is higher than less experted ones; more precisely a fixed cost is associated to each patrol and it is activated only if the patrol is scheduled in the plan. Patrols are grouped in classes of experience and fixed costs are homogeneous within each class of experience. A limited number of patrols for each class is available.Each route start from the company station and must come back to the station before the end of the working period (generally patrols work in the period 10 p.m.-6 a.m.). A unitary distance cost, equal for all the patrols, and a unitary time cost, varying among classes of patrols, are defined. The objective function consist into minimizing a generalized cost function which is obtained as a linear combination of distance cost, time cost and patrol activation cost, to which a penalty is added for each spacing constraint and time windows violation.The resulting problem is a novel challenging highly constrained VRP, in which different extensions of the VRP are combined with newly introduced type of constraints. In fact, this problem is an extension of the Periodic Vehicle Routing Problem with Time Windows, in which a further spacing constraint between consecutive visits is added. In fact, differently from what occurs in classical periodic routing problems, in the SPVRP it is not fair to schedule a visit at the end of one period and another one at the beginning of the following period, because of the minimum spacing constraint. Thus, in SPVRP, it is important not only the period in which a customer is served but also the moment within the period in which the service is performed. Another important difference is that, in classical Periodic VRPs each route is performed within a unique time period while in the SPVRP routes may be performed over different periods.More over, an heterogeneous fleet of vehicles is considered, but differently from classical VRP with heterogeneous fleet, the different classes of vehicles does not differ among each others only from different fixed and/or variable usage costs, but they also have different restrictions on the checkpoints which may be visited with each class of vehicles (in this case, each class of patrols).Furthermore, different service times at nodes are considered and an additional constraint on the maximum route duration length is addressed. The final goal of the surveillance company is not only to schedule a fair routing plan at the minimum cost, but also to define a set of potential high quality plans from which to chose each day a different plan to be followed. This request came out from the need of avoiding to repeat always the same tours, for a security reason. In fact, if criminals know exactly at what time a patrol visit a checkpoint, they may easily elude surveillance. More unpredictable is the visits schedule, more difficult is to elude surveillance, with a consequent increase of the public safety level and a reduction of committed crimes. For this reason a second optimization problem may be defined, consisting into choosing a subset of K solutions within a set of N feasible solutions, in order to minimize a generalized cost function, which is given by a weighted sum of the average solutions cost, and of repetitiveness penalty costs. In fact, two repetitiveness indexes are defined. The first one, deals with visit times at checkpoints and works as follows. If, a checkpoint is visited within the same timeslot in more than p% of the solutions belonging to the subset, a visit time repetitiveness penalty is added. A similar procedure is applied for sequence repetitiveness. In fact, if a checkpoint is visited immediately after the same checkpoint in more than q% of the solutions in the subset, a sequence penalty is added. The value of p and q are parameters of the algorithm. Timeslots are defined as very small consecutive disjointed time intervals (i.e. 10 minutes). The goal of the problem is to find the subset which minimizes the generalized cost function.

2 A GRASP for the SPVRP

The greedy randomized adaptive search procedure (also known as GRASP) is a metaheuristic algorithm commonly applied to combinatorial optimization problems. GRASP typically consists of iterations made up from successive constructions of a greedy randomized solution and subsequent iterative improvements of it through a local search. The greedy randomized solutions are generated by adding elements to the problem's solution set from a list of elements ranked by a greedy function according to the quality of the solution they will achieve. To obtain variability in the candidate set of greedy solutions. Since in this problem we are looking for a pool of good quality solutions, and not only for the best solution, GRASP fit well for this purpose.

The GRASP proposed for the SPVRP works as follows. At each iteration of the algorithm, a greedy solution is constructed. At each step of the construction phase, next checkpoint to be visited is randomly draw according distance based probability, i.e. the probability of each checkpoint to be chosen as next, is based on the inverse of the distance of the checkpoint and the last visited one (or, when the first customer of a route must be selected, based on the inverse of the distance of the checkpoint and the station). If a checkpoint has been already visited by the same route or by another one, during the current time window, or within a fixed minimum spacing time, its probability to be chosen is put equal to zero. For instance, assume that checkpoint a must be visited once during the time window 12 p.m.-2 a.m. and once during the time window 2 a.m.-4 a.m and the minimum spacing time between visits has been fixed equal to 60 minutes; if a has been visited once at 1:30 a.m. and if inserted in the current route at the current point it would be visited at 2:10 a.m. its probability to be chosen at this point is forced to zero, because, even if the two visits would respects time windows constraints, the minimum required spacing between consecutive visits would not be respected. A route is closed when no more available customers may be inserted in or maximum duration is reached. After a route is closed, if there are still unrouted customers. Once a feasible solution is constructed, a local search is applied consisting into destroying almost empty routes and try to relocate the unrouted customers within the remaining routes at the minimum generalized cost. This overall procedure is repeated for a large number of times in order to create a pool of good quality solution from which we need to chose the best subset. To this a randomized search procedure is developed in which at each iteration K solutions are picked up basing on a probability inversely proportional to their generalized cost. The procedure is repeated L times and at the end, the best subset (where the quality of the subset is given by a weighted sum of average cost and repetitiveness penalties) is chosen. Computational results on very large realistic instances will be reported at the conference.