Inventory rebalancing in bike-sharing systems

Marco Casazza[1], Alberto Ceselli[1], and Roberto Wolfler Calvo[2]

[1]Università degli Studi di Milano, Dipartimento di Informatica, Italy.
[2]Université Paris 13, Laboratoire d'Informatique Paris Nord, France.

We address an optimization problem arising in rebalancing operations of inventory levels in bike-sharing systems. Such systems are public services where bikes are available for shared use on a short term basis. To ensure the availability of bikes in each station and avoid disservices, the bike inventory level of each station must met a forecast value. This is achieved through the use of a fleet of vehicles moving bikes between stations. Our problem can be classified as a Split Pickup and Split Delivery Vehicle Routing Problem. We propose a formulation in which routes are decomposed in smaller structures and we exploit properties on the structure of the optimal solutions, to design an exact algorithm based on branch-and-price.

1 Introduction

A bike-sharing system is a public service where bikes are available for shared use to individuals on a short term basis. Typically, users can pick up bikes at a cost and drop them back at designated stations widespread around the city. Such systems have been implemented around the world and are now present in hundreds of cities, as documented in [1].

Indeed the organization and management of the logistics of a bike-sharing system is challenging: for example, user behaviour results in an imbalance of the bike inventory in the stations over time, leading to undesired disservices such as empty departure stations or full destinations. To ensure the availability of service, one of the solutions chosen by many operators is to rebalance the bike inventory level of the stations by means of a fleet of dedicated trucks: the target inventory level of each station is forecast, and bikes are picked up from stations where congestion is expected, and delivered to those expected to become empty. Due to the high costs of running trucks in a urban environment, efficient rebalancing operations are a key factor for the success of the whole system.

Unfortunately, such operations rise very hard optimization problems. That is the case of the *Split Pickup and Split Delivery Vehicle Routing Problem (SPSDVRP)*: we assume that the network of the bike stations is given with both travel time and cost between each pair of stations. For each station we also know both the current and the target bike inventory levels. A fleet of homogeneous vehicles of limited capacity is given to perform rebalancing operations in such a way that the target inventory level is met for each station of the network. However, during the rebalancing process each station can be visited multiple times, even by the same vehicle. The SPSDVRP requires to find a route for each vehicle, that is a pattern defining which stations are visited, the order of visits, and the amount of bikes loaded or unloaded at each station. Each vehicle always starts and ends at a depot with no bikes on board. Furthermore, each route cannot exceed a given time limit, that is the operator shift duration. We finally assume that no station is used as a temporary unloading location, meaning that during rebalancing operations no bikes are loaded from a delivery station or unloaded to a pickup station and therefore the number of bikes in each station is monotone. A feasible solution to the SPSDVRP consists of a set of routes respecting the above conditions. A solution is also optimal when the sum of the travelling costs of all vehicles is minimum.

From a methodological point of view, our SPSDVRP is NP-hard and belongs to the wide class of *Pickup and Delivery Vehicle Routing Problems (PDVRPs)* [2] and generalises the *Split Delivery Vehicle routing Problem (SDVRP)*. The problem of rebalancing bikes with a single vehicle has been addressed in [3], while a first mathematical approach for the SPSDVRP on a bike-sharing system has been proposed in [4]: the authors model the problem as a set partitioning extended formulation in which each variable represents a full vehicle route. However, such a formulation has two main drawbacks: first, it is not designed to handle travelling times, that are instead approximated by a limit on the number of visits in each route, and second, solving the continuous relaxation by means of column generation techniques is time consuming due to the complexity of the pricing procedure. We propose instead a new formulation that overcomes these two limitations, new theoretical properties on the structure of the optimal solutions, and a branch-and-price approach that solves to optimality instances with up to 20 nodes.

2 Model

The SPSDVRP on a bike-sharing system can be formalized as follows: a set of station nodes $N = \{1 \dots n\}$ is given, each with both the current and the target bike inventory levels $stock_i$ and $target_i$, respectively. When $stock_i > target_i$ we say that i is a *pickup* node, when $stock_i < target_i$ it is a *delivery* node, and when $stock_i = target_i$ i it is already balanced. The demand of each node $d_i = |stock_i - target_i|$ is the quantity of bikes to pick up from (resp. deliver to) that node.

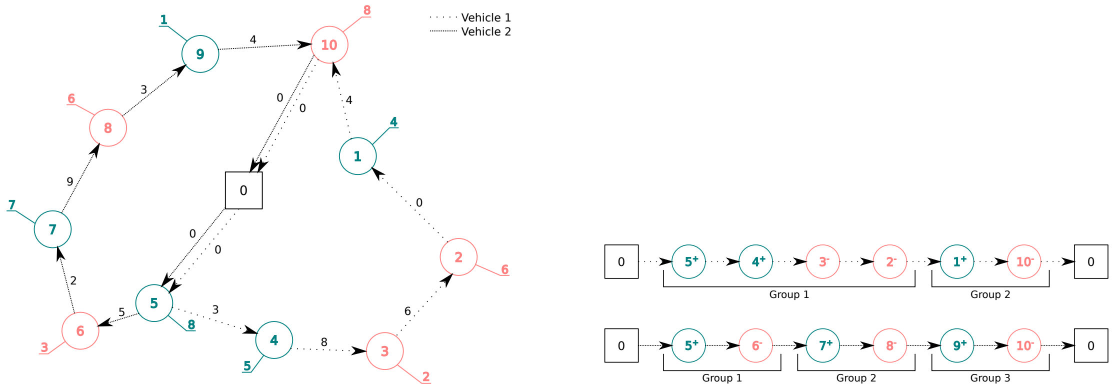
Let $G = (N_0, A)$ be a directed graph in which $N_0 = N \cup \{0\}$ is the set of nodes including the depot 0, and $A = \{(i, j) \mid i, j \in N_0\}$ is the set of arcs. Let c_{ij} and t_{ij} be the travelling cost and time of arc $(i, j) \in A$, respectively. W.l.o.g we assume that both costs and times satisfy triangular inequality.

A homogeneous fleet of vehicles $M = \{1 \dots m\}$ is given to satisfy station node demands. Each vehicle has a capacity C and a time resource T.

The SPSDVRP on a bike-sharing system is the problem of redistributing bikes in the network at minimum travelling cost, satisfying node demands while not exceeding neither vehicles capacity nor their time resource. Pickup and delivery nodes can be visited several times, either by the same vehicle or by different ones, and therefore their demand can be split.

3 Groups formulation and properties

The approach to the SPSDVRP proposed in [4] revealed that solving the continuous relaxation of such formulation was very challenging due to the structure of the pricing problem. That



(a) Example of solution on a graph having 10 nodes and 2 vehicles: each nodes is either a pickup ($^+$) or a delivery ($^-$). At each node it is attached a label reporting its demand. Each arc traelled by a vehicle has a label indicating the current amount of bikes on board.

(b) Group partitioning of the routes in solution in Figure 1(a): for each vehicle, each sequence of pickups and delivieries corresponds to a single group. Groups are concatenated to compose a route.

Figure 1: Example of SPSDVRP solution and its group representation.

motivated us to elaborate on a different approach, identifying particular regularities and properties of combinatorial substructures of the routes, and trying to reduce the complexity of the pricing problem by exploiting these properties.

We start from a simple observation:

Observation 1. *A route always starts with a sequence composed only by pickup nodes, always ends with a sequence composed only by delivery nodes, and in general always interleaves sequences of pickups followed by sequences of deliveries.*

Our intuition is therefore that the structure of a route can be much simplified by explicitly encoding such an interleaved behaviour. We formalize such an intuition denoting as *group* a sequence of one or more pickup nodes followed by a sequence of one or more delivery nodes and define a route as a sequence of concatenated groups. We then express a route cost and time in function of its groups. An example of routes and groups is depicted in Figure 1(b).

We then provide proofs that:

Theorem 1. *There always exists an optimal solution in which no node is visited more than once in the same group.*

Theorem 2. *For each pair of pickup (resp. delivery) nodes, there always exists an optimal solution in which they are visited at most once in the same group.*

Theorem 3. *Given the sets of nodes visited in each group of each vehicle, the problem of assigning the quantity loaded (resp. unloaded) at each station can be solved in polynomial time.*

4 Algorithms

We model our SPSDVRP as the problem of finding a minimum cost set of groups concatenations. Our formulation has an exponential number of variables, one for each group of each vehicle, and we recur to column generation techniques to solve its continuous relaxation.

Our pricing problem is a variant of *Resource Constrained Elementary Shortest Path Problem (RCESPP)* on a particular graph in which pickup nodes must be visited before delivery ones, and the profit collected at each node may be fractional. However, concerning the pricing problem we prove that:

Theorem 4. *There always exists an optimal solution in which there is at most one fractional pickup node (resp. delivery node) in the same group.*

Theorem 5. *The pickup (resp. delivery) node selected as fractional in each group, has a profit that is not greater than the less profitable non-fractional node visited in the same group.*

We solve our pricing problem by means of a label correcting algorithm in which each label is extended in three different ways: one collecting full profit, one collecting fractional profit, and one for when it is profitable to visit the node only without collecting any profit.

We included our CG procedure into a branch-and-price framework to achieve integrality of solutions, with heuristic pricers to speed-up CG, and a rounding heuristic to obtain good upper bounds to the value of the optimal solutions. We implemented our algorithms in C++ using SCIP framework and tested our methodology against datasets from the literature [4]. The results shown in Table 1 are promising: our algorithm solves to proven optimality all instances with 10 nodes in an average computing time smaller than 1 minute.

Instance	$\underline{z}$	$\bar{z}$	gap(%)	nodes	time(s)	Instance	$\underline{z}$	$\bar{z}$	gap(%)	nodes	time(s)
n10q10a	3719.00	3719	0.00	130	8.92	n10q10f	3037.00	3037	0.00	57	2.57
n10q10A	3055.00	3055	0.00	53	2.67	n10q10F	4097.00	4097	0.00	103	3.15
n10q10b	3353.00	3353	0.00	984	37.52	n10q10g	4179.00	4179	0.00	237	7.50
n10q10B	3745.00	3745	0.00	99	5.28	n10q10G	4221.00	4221	0.00	67	5.06
n10q10c	4239.00	4239	0.00	5	1.17	n10q10h	4194.00	4194	0.00	396	10.98
n10q10C	3392.00	3392	0.00	307	15.12	n10q10H	4118.00	4118	0.00	640	29.38
n10q10d	4472.00	4472	0.00	153	7.85	n10q10i	2523.00	2523	0.00	161	6.23
n10q10D	3307.00	3307	0.00	2771	72.46	n10q10I	3287.00	3287	0.00	72	2.16
n10q10e	3816.00	3816	0.00	870	54.63	n10q10j	3343.00	3343	0.00	150	6.76
n10q10E	4876.00	4876	0.00	154	5.22	n10q10J	3161.00	3161	0.00	169	3.78

Table 1: Results for instances with 10 nodes.

References

[1] P. DeMaio, *Bike-sharing: History, impacts, models of provision, and future*, Journal of Public Transportation, 12(4):4156, 2009.

[2] M. Battarra, J.-F. Cordeau, and M. Iori. *Pickup and delivery problems for goods transportation*, in Vehicle Routing: Problems, Methods, and Applications, Second Edition, chapter 6, pages 161–191, SIAM, 2014.

[3] D. Chemla, F. Meunier, R. Wolfler Calvo, *Bike sharing systems: Solving the static rebalancing problem*, Discrete Optimization, Volume 10, Issue 2, May 2013, Pages 120–146.

[4] M. Casazza, A. Ceselli, D. Chemla, F. Meunier, R. Wolfler Calvo, *The Multiple Vehicle Balancing Problem*, technical report, 2016, https://hal.archives-ouvertes.fr/hal-01351215.