The Static Bicycle Repositioning Problem - Literature Survey and New Formulation

Hans Martin Espegren, Johannes Kristianslund, Henrik Andersson, and Kjetil Fagerholt

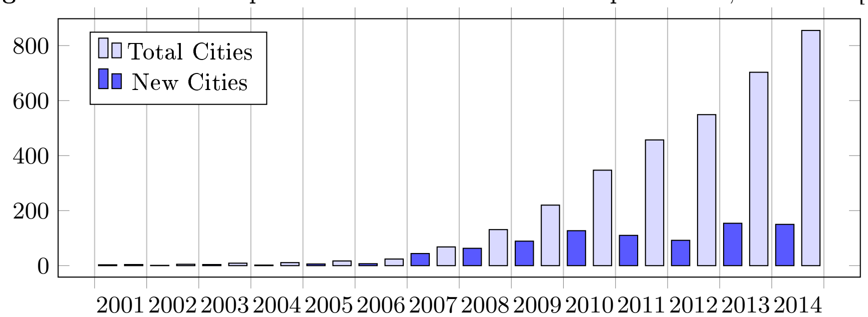
Department of Industrial Economics and Technology Management, Norwegian University of Science and Technology, Trondheim, Norway

Abstract. This paper considers the static bicycle repositioning problem (SBRP), which deals with optimally re-balancing bike sharing systems (BSS) overnight, i.e. using service vehicles to move bikes from (nearly) full stations to (nearly) empty stations. An exhaustive literature survey comparing existing models is presented, and a new and improved mathematical formulation for the SBRP is proposed. The model is tested on a number of instances generated based on data from a real BSS.

1 Introduction

As urbanization proceeds throughout the world, public decision makers are looking for effective, affordable, and environmentally friendly means of transportation. Bike sharing fulfills these criteria for short distance traveling within city centres, and consequently bike sharing is getting increased attention from both governments and the public. Currently there are 948 cities with an active Bike Sharing System (BSS) and 273 with a system under planning or construction [10]. Figure 1 shows the expansion of bike sharing the recent years, expressed as number of cities in the world with a public BSS. For an extensive review of the historical development of BSSs, the reader is referred to [9], [35], and [24].

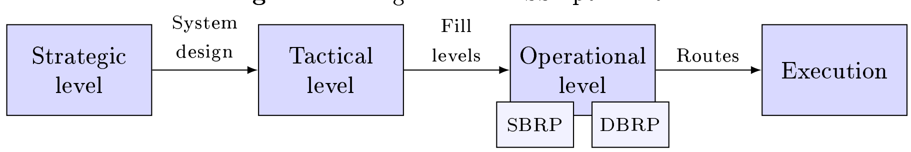
Fig. 1. Worldwide development in number of cities with a public BSS, 2000-2014 [10]



Bike sharing is a public system for automatic or semi-automatic lending of bicycles for use within a restricted time period and area. A bike can be lent at one station and delivered at another. Note that during the night most systems are either closed or in limited use. For the system to function well, it is crucial that there are bikes available at a station when someone wants to pick up a bike and that there are free slots available when someone wants to return one. To achieve this, most BSSs use service vehicles to re-balance the system, i.e. to move bikes from (nearly) full stations to (nearly) empty stations. This paper studies one important aspect of the operation of BSSs, namely the logistics of the service vehicles used to re-balance the system overnight.

The planning problems arising from BSSs are divided into three levels in accordance with [37]; a strategic, a tactical, and an operational level, as illustrated in Figure 2. The strategic level contains problems that arise when designing the system, e.g. determining the optimal number of bikes and locations of stations. On the tactical level the objective is to find an optimal distribution of bikes between the stations at a specific time, while finding optimal routes for the service vehicles to re-balance the system is the objective at the operational level.

Fig. 2. Planning levels of BSS optimization



It is common to divide the operational level in two: static and dynamic problems. In line with [31], the problems are named *static bicycle repositioning problem* (SBRP) and *dynamic bicycle repositioning problem* (DBRP). The SBRP is typically used for overnight balancing, when the demand forecast for the operating period is not considered; the problem is static and deterministic. To describe the SBRP we introduce the concept of states, i.e. a distribution of bikes throughout the system, expressed as a specific number of bikes at each station. The *optimal state* is the desired distribution of bikes at the end of the planning period, i.e. early in the morning, while the *initial state* is the distribution at the beginning of the planning period, i.e. late in the evening. After solving the model, we get the *final state*. The difference between the final state and optimal state is called deviation. All stations and vehicles have restricted capacities, and the fleet of service vehicles may be either homogeneous or heterogeneous. For every vehicle, a complete route and the number of bikes to pick up or deliver at each station must be decided. Hence the SBRP can be classified as a *static many-to-many one-commodity pickup and delivery problem with selective pickups and selective deliveries*, in accordance with [3]. The DBRP is on the other hand used for intraday re-balancing, as the demand during the operating time is taken into account. Hence, the DBRP is both dynamic and stochastic.

In this paper we focus on the SBRP. In the literature survey, we identify a need for a new formulation of the problem including more real-life aspects important for system planners. Our contributions are 1) to present an exhaustive literature survey on the SBRP, including a systematic comparison of the existing models, and 2) to propose a new mathematical model of the problem that captures more real-life aspects. We also propose symmetry-breaking constraints and valid inequalities to tighten the formulation. The model is tested on a number of test instances based on data from a real BSS.

Section 2 provides the literature survey on the SBRP, while a new mathematical model for the SBRP is introduced in Section 3. A computational study is presented in Section 4 and concluding remarks are given in Section 5.

2 Literature Survey

In this literature survey we focus on the static bicycle repositioning problem (SBRP). For studies on the strategic level, we refer to [16], [23], and [33] that determine the number of stations and their locations, and to [15] that finds the optimal number of bikes in the system and the number of slots at each station. At the tactical level we can refer to [30], [34], and [38] for analyses of the placement of bikes, while [20] studies the detection of broken bikes in the system. There are also a number of studies regarding the DBRP, see for example [1], [4], [5], [7], [21], [26], [27], [32], and [39]. An overview of planning problems arising in shared mobility systems, for example a bike-sharing system, is given in [22].

The SBRP was first studied in [2]. They describe the system using graph-theory. The objective is to move bikes along the arcs so each station is perfectly re-balanced at minimal cost. One of the main findings is that the SBRP is NP-hard. In [6], the work from [2] is continued. An optimization model is presented, but shows to be hard to solve, so they relax the problem by removing the sequential dimension and solve it using a branch-and-cut (B&C) algorithm.

In [31], two different mixed integer programming formulations are introduced; an arc-indexed and a time-indexed. The objective is to minimize a weighted sum of the stations' penalty costs for deviations and the operating cost. The authors conclude that the arc-indexed model provides the best results for most instances, but the time-indexed formulation is easier to adapt to the DBRP. Valid inequalities and dominance rules are proposed to strengthen the formulations.

The arc-indexed formulation from [31] is enhanced in [19] and [14], both proposing methods for solving larger instances. In [19], the formulation is simplified by allowing only one vehicle, stating that a station is either a pickup or delivery station and assuming that each station only can be visited once. The objective is to minimize a penalty function depending on the number of bikes at each station. The authors present a construction heuristic used to generate an initial solution followed by a tabu search. On the other hand, the model is expanded in [14] by using a three-step algorithm. In the first step, stations are clustered using a saving heuristic. In the second step, vehicles are assigned to clusters, while the routes for each vehicle are determined in the third step.

The SBRP is represented using a complete directed graph in [28] and [29]. Further, several metaheuristics are presented and tested. The authors conclude that Variable Neighborhood Search (VNS) yields the best results on instances of moderate size, while a PILOT/GRASP hybrid turns out to be superior on large instances. A neighborhood search is also used in [17]. Two formulations for the SBRP are also developed; a routing model and a step model, both incorporated in a Large Neighborhood Search (LNS). The routing model uses an arc-indexed formulation, while the step model allocates all station visits to routes.

In [34], the SBRP is solved in combination with the tactical level problem of finding the optimal states. The routes from the SBRP must satisfy the service level requirements from an inventory problem. The objective is to minimize the maximal route length, hence it is formulated as a makespan problem. To solve the model the authors propose a cluster first route second heuristic.

Four possible formulations of the SBRP are tested and discussed in [8]. To handle the exponential number of subtour eliminating constraints, a B&C algorithm is proposed in addition to both valid inequalities and separation procedures. The authors conclude that the subtour elimination and separation techniques proposed by [18] for the 1-PDTSP give the best computational results.

A decomposition method is introduced in [36], consisting of a request generation algorithm and a *bike request scheduling problem* (BRSP). The request generation algorithm uses various data to generate repositioning requests. A request includes the location and number of bikes to be picked up or delivered, a time window and an importance weight. The BRSP determines which requests to execute and assigns them to vehicles. The objective is to minimize the total weight of rejected requests.

The objective of the SBRP-model in [27] is to maximize the number of rebalanced stations, only allowing pickup and delivery of full truckloads of bikes. The authors use a heuristic that solves the one-vehicle problem for each vehicle.

In [13], the SBRP is decomposed using a Benders decomposition scheme. The subproblem determines the pickup and delivery quantities along a fixed route of station visits, while the master problem finds new routes visiting all stations with too few or too many bikes. In a later study, [12], the authors use insights from [13] to solve the SBRP formulation from [6]. Whilst [6] could only find heuristic solutions for realistically sized instances, the method from [12] yield optimal solutions.

Table 1 shows a comparison of the main characteristics of the SBRP models in the studies surveyed above, as well as some key information about the solution methods. Note that the mathematical model proposed in Section 3 is also included in the table. The numbers in the top row correspond to the numbers in Table 2.

From the table it becomes evident that half of the studies solve the problem with only one service vehicle, even though most problems of realistic size use several. Note that many articles use clustering algorithms. By assigning each cluster to a vehicle, the SBRP could be solved once for each vehicle. Among the studies allowing multiple vehicles, two assume the fleet to be homogeneous. Half of the studies allow multiple visits to a station, while the other half does not. When the deviation between the optimal and initial state is larger than the vehicle capacity, allowing multiple visits to each station seems most reasonable.

Table 1. Comparison of SBRP Articles. The numbers in the table header are explained in Table 2.

Parameter	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Multiple vehicles	No	No	Yes	Yes	Yes	No	No	Yes	Yes	No	No	Yes	Yes	Yes
Heterogeneous fleet	n/a	n/a	Yes	Yes	Yes	n/a	n/a	No	Yes	n/a	n/a	No	Yes	Yes
One vehicle can visit a station multiple times	Yes	Yes	No/Yes two mod.	Yes	Yes	No	No	No	Yes	No	Yes	No	No	Yes
Several vehicles can visit the same station	n/a	n/a	Yes	Yes	Yes	n/a	n/a	No	Yes	n/a	n/a	No	No	Yes
Modeling loading/unloading	No cost for l/unl.	No cost for l/unl.	Depends on quantity	Average time for station	No cost for l/unl.	Depends on quantity	No cost for l/unl.	Time for each request	Average time for station	Depends on quantity	No cost for l/unl.	Average time for station	Depends on quantity	Depends on quantity
Selective pickup/delivery	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes
Stations used as temporal inventories	No	Yes	Yes	No	Yes	No	No	No	Yes	No	Yes	No	Yes	No
Allows non-perfect rebalancing	No	No	Yes	Yes	No	Yes	No	Yes	Yes	No	No	Yes	Yes	Yes
Objective function	Min. cost	Min. cost	Min. dev. and cost	Min. dev & operations	Min. longest route	Min. penalty func.	Min. cost	Min. rejected requests	Min. dev. and cost	Min. time and cost	Min. cost	Max. rebal. nodes	Min. dev. and cost	Min. dev. and time
Subtour elimination	Many new var. & cons.	Many new var. & cons.	MTZ / Time	Heuristics	Time	MTZ	MTZ / Sep. & cut	Time	Time / Route	Separation & cut	Separation & cut	A variant of MTZ	MTZ	Strengthened MTZ
Solution method	n/a	B&C & Tabu search	Heuristics & exact	PILOT, GRASP, VNS	Cluster first route second	Tabu search	Exact: B&C	n/a	LNS	Exact: Benders decomp.	Exact: Benders cut	Heuristics & exact	Cluster first route second	Exact
Size of solvable instances	n/a	1v., 100st.	1v., 60st.	21v., 700st.	5v., 135st.	1v., 400st.	1v., 50st.	n/a	6v., 240st.	1v., 50st.	1v., 60st.	5v., 100st.	3v., 200st.	2v., 15st.
Based on article (table header as reference)	n/a	1	n/a	n/a	n/a	3	n/a	n/a	4	n/a	2 & 10	n/a	3	n/a

Table 2. Articles overview for Table 1

1	Benchimol et al. [2]	**8**	Sörensen and Dilip [36]
2	Chemla et al. [6]	**9**	Gaspero et al. [17]
3	Raviv et al. [31]	**10**	Erdoğan et al. [13]
4	Rainer-Harbach et al. [28] & [29]	**11**	Erdoğan et al. [12]
5	Schuijbroek et al. [34]	**12**	O'Mahony and Shmoys [27]
6	Ho and Szeto [19]	**13**	Forma et al. [14]
7	Dell'Amico et al. [8]	**14**	Espegren et al. (this study)

Five studies assume that there is no time usage or cost associated with the loading and unloading operations at the stations, three use an average time and five studies let the time usage depend on the number of bikes handled. Note that none of the studies take traffic congestion into account, but presume the driving time between two stations to be constant. Just one study, [27], allows only full truckloads.

The studies by [2], [6], [34], [8], [13], and [12] minimize the time and/or cost associated with repositioning the bikes. In these studies, the solutions are only valid if the number of deviations is zero, i.e. the system is perfectly re-balanced. The remaining studies use objective functions that in various ways minimize the number of deviations.

All but two studies ([2] and [36]) include computational experiments on either theoretical or real instances. The majority use some kind of heuristics to solve the instances. All studies that use exact methods fail to find the optimal solution when the problem size increases and only yield upper and lower bounds. Since the problems include binary and/or integer variables, a common approach is to use B&C algorithms. The cuts can be generated using inequalities from [18] or using Benders decomposition [12]. Popular heuristics are tabu search and VNS/LNS. In [14] the problem is decomposed, and one part is solved by a heuristic and another part using exact methods.

The studies using a time-variable do not need subtour eliminating constraints. Among the remaining articles, the MTZ-formulation [25] is widely used to avoid subtours, while three studies, [8], [13], and [12], eliminate subtours using separation algorithms and cuts.

3 Mathematical Formulation

In this section, we propose a new mathematical model for the SBRP. The objective of the model is to minimize a weighted combination of the total deviation in the number of bikes at each station from the optimal state at the time limit and the time used. We assume a heterogeneous fleet of service vehicles that start and finish their routes empty at the depot. Several vehicles can visit the same station and a single vehicle can visit the same station several times. We presume the driving time between stations to be constant and independent of the hour. In addition to the driving time, each vehicle uses a fixed parking time at each station visit. Time used to load and unload bikes at a station is proportional to the number of bikes handled plus a given parking time. All stations are defined as either *pickup stations* or *delivery stations* depending on their initial state relative to their optimal state. It is not possible to pick up bicycles at a delivery station or deliver them at pickup station.

Each station $i \in \mathcal{N}$ has a set of possible visits $\mathcal{M}_i$. Note that the depot is included in this set. Our formulation uses arc flow variables x_{imjnv}, $i \in \mathcal{N}$, $m \in \mathcal{M}_i$, $j \in \mathcal{N}$, $n \in \mathcal{M}_j$, $v \in \mathcal{V}$ indicating whether vehicle v drives from station visit (i,m) to station visit (j,n) or not, where m and n are the station visit numbers. The entire notation is presented in Table 3.

Table 3. Notation used in the mathematical formulation

Sets	
$\mathcal{N}$	Set of stations, indexed by i, j
$\mathcal{V}$	Set of vehicles, indexed by v
$\mathcal{M}_i$	Set of possible visits at station i, indexed by m, n
Parameters	
T_{ij}^D	Driving time between stations i and j
T^P	Time used for parking a vehicle
T^H	Handling time used for loading or unloading a bike
$\overline{T}$	Time limit for operation of service vehicles
Q_v	Capacity of vehicle v
J_i	1 if station i is a pickup station, and -1 if it is a delivery station
α	Weight on deviations in the objective function relative to time usage
$\overline{A}$	Maximum number of station visits for a vehicle
I_i	Initial state, number of bikes at station i
O_i	Optimal state, number of bikes at station i
Variables	
x_{imjnv}	1 if vehicle v is driving directly from station visit (i, m) to station visit (j, n), 0 otherwise
f_{ijv}	Total number of bikes carried by vehicle v between stations i and j
q_{iv}	Number of bikes either picked up or delivered at station i by vehicle v
y_i	Final state, number of bikes at station i
u_{imv}	The sequence number in which station visit (i, m) is made by vehicle v

$$\begin{aligned} \min \quad & \alpha \sum_{i \in \mathcal{N}} J_i (y_i - O_i) \\ & + (1-\alpha) \left[\sum_{i \in \mathcal{N}} \sum_{m \in \mathcal{M}_i} \sum_{j \in \mathcal{N}} \sum_{n \in \mathcal{M}_j} \sum_{v \in \mathcal{V}} \left(T_{ij}^D + T^P \right) x_{imjnv} + \sum_{i \in \mathcal{N}} \sum_{v \in \mathcal{V}} T^H q_{iv} \right] \end{aligned} \tag{1}$$

subject to:

$$\sum_{j \in \mathcal{N}} \sum_{n \in \mathcal{M}_j} x_{dvjnv} = 1 \qquad v \in \mathcal{V} \tag{2}$$

$$\sum_{i \in \mathcal{N}} \sum_{m \in \mathcal{M}_i} x_{imd(v+|\mathcal{V}|)v} = 1 \qquad v \in \mathcal{V} \tag{3}$$

$$\sum_{j \in \mathcal{N}} \sum_{n \in \mathcal{M}_j} x_{jnimv} - \sum_{j \in \mathcal{N}} \sum_{n \in \mathcal{M}_j} x_{imjnv} = 0 \qquad i \in \mathcal{N} \setminus \{d\}, m \in \mathcal{M}_i, v \in \mathcal{V} \tag{4}$$

$$\sum_{j \in \mathcal{N}} \sum_{n \in \mathcal{M}_j} \sum_{v \in \mathcal{V}} x_{imjnv} \leq 1 \qquad i \in \mathcal{N}, m \in \mathcal{M}_i \tag{5}$$

$$\sum_{j \in \mathcal{N}} f_{jiv} + J_i q_{iv} - \sum_{j \in \mathcal{N}} f_{ijv} = 0 \qquad i \in \mathcal{N}, v \in \mathcal{V} \tag{6}$$

$$y_i + \sum_{v \in \mathcal{V}} J_i q_{iv} = I_i \qquad i \in \mathcal{N} \tag{7}$$

$$\sum_{v \in \mathcal{V}} q_{iv} - J_i(I_i - O_i) \leq 0 \qquad i \in \mathcal{N} \tag{8}$$

$$f_{ijv} - \sum_{m \in \mathcal{M}_i} \sum_{n \in \mathcal{M}_j} Q_v x_{imjnv} \leq 0 \qquad i, j \in \mathcal{N}, v \in \mathcal{V} \tag{9}$$

$$\sum_{j \in \mathcal{N}} f_{djv} = 0 \qquad v \in \mathcal{V} \tag{10}$$

$$\sum_{i \in \mathcal{N}} f_{idv} = 0 \qquad v \in \mathcal{V} \tag{11}$$

$$\sum_{i \in \mathcal{N}} \sum_{m \in \mathcal{M}_i} \sum_{j \in \mathcal{N}} \sum_{n \in \mathcal{M}_j} (T_{ij}^D + T^P) x_{imjnv} + \sum_{i \in \mathcal{N}} T^H q_{iv} \leq \overline{T} \qquad v \in \mathcal{V} \tag{12}$$

$$u_{imv} - u_{jnv} + (\overline{A} - 1) x_{imjnv} + (\overline{A} - 3) x_{jnimv} \leq \overline{A} - 2 \qquad i, j \in \mathcal{N}, m \in \mathcal{M}_i, n \in \mathcal{M}_j, v \in \mathcal{V} \tag{13}$$

$$x_{imjnv} \in \{0, 1\} \qquad i, j \in \mathcal{N}, m \in \mathcal{M}_i, n \in \mathcal{M}_j, v \in \mathcal{V} \tag{14}$$

$$f_{ijv} \geq 0, \text{integer} \qquad i, j \in \mathcal{N}, v \in \mathcal{V} \tag{15}$$

$$q_{iv} \geq 0, \text{integer} \qquad i \in \mathcal{N}, v \in \mathcal{V} \tag{16}$$

$$y_i \geq 0, \text{integer} \qquad i \in \mathcal{N} \tag{17}$$

$$u_{imv} \geq 0, \text{integer} \qquad i \in \mathcal{N}, m \in \mathcal{M}_i, v \in \mathcal{V} \tag{18}$$

The objective function (1) consists of two terms that are to be minimized. The first term is the deviation in number of bikes between the final state, y_i, and the optimal state, O_i, for all stations. Having too many and too few bikes are equally penalized. The second term is the total time used to obtain the final state. Total time corresponds to the sum of driving time, T_{ij}^D, parking time, T^P, and handling time, T^H. By setting α slightly below one, the most effective routes minimizing the deviation are found.

Constraints (2) and (3) force the vehicles to start and end at the depot, d. Symmetry at the depot is handled by stating that vehicle v uses visit numbers v and $v + |\mathcal{V}|$ when leaving and arriving at the depot, respectively. Constraints (4) ensure that a vehicle that enters a station visit, leaves the same station visit, while constraints (5) make sure all station visits happen at most once.

The loading and unloading constraints (6) ensure that the flow of bikes into station i, f_{jiv}, equals the flow out of the station, f_{ijv}, plus the net pickup, q_{iv}. Since the problem is static, only the total net pickup is considered. Constraints

(7) and (8) assign values to the final state, y_i. In addition, constraints (8) give an upper bound on the net pickup at station i by vehicle v, q_{iv}.

The vehicle capacity constraints (9) make sure that a vehicle never carries more bikes along an arc than the vehicle's capacity multiplied by the number of times the arc is traversed. Constraints (10) and (11) state that the service vehicles must be empty when leaving and returning to the depot. Capacity constraints for the stations are handled implicitly. The total time spent for each vehicle is limited to $\overline{T}$ by constraints (12).

Subtours are handled in constraints (13), similar to the Miller-Tucker-Zemlin (MTZ) constraints [25], but with a strengthening proposed in [11]. Various methods for eliminating subtours have been tested, and these constraints showed to perform best.

Symmetry breaking constraints remove solutions that are mathematically different, but practically identical, while adding valid inequalities is a way of improving the solution of the linear relaxation. Various symmetry breaking constraints and valid inequalities have been tested, and the ones presented here are those found most effective.

$$\sum_{j \in \mathcal{N}} \sum_{n \in \mathcal{M}_j} \sum_{v \in \mathcal{V}} \left(x_{imjnv} - x_{i(m-1)jnv} \right) \leq 0 \quad i \in \mathcal{N} \setminus \{d\}, m \in \mathcal{M}_i \setminus \{1\} \tag{19}$$

$$\begin{aligned} \sum_{i \in \mathcal{N}} \sum_{m \in \mathcal{M}_i} \sum_{j \in \mathcal{N}} \sum_{n \in \mathcal{M}_j} \left(T_{ij}^D + T^P \right) \left(x_{imjnv} - x_{imjn(v+1)} \right) & \\ + \sum_{i \in \mathcal{N}} T^H \left(q_{iv} - q_{i(v+1)v} \right) \geq 0 \qquad v \in \mathcal{V} \setminus \{|\mathcal{V}|\} \; \Big| \; Q_v = Q_{(v+1)} & \end{aligned} \tag{20}$$

Constraints (19) reduce symmetry by handling the station visits, so that they appear in the right sequence. By introducing constraints (20), symmetry that occurs when using a homogeneous fleet of service vehicles is reduced.

$$\sum_{v \in \mathcal{V}} q_{iv} - |\,(I_i - O_i)\,| \sum_{m \in \mathcal{M}_i} \sum_{j \in \mathcal{N}} \sum_{n \in \mathcal{M}_j} \sum_{v \in \mathcal{V}} x_{imjnv} \leq 0 \qquad i \in \mathcal{N} \tag{21}$$

$$\sum_{v \in \mathcal{V}} \sum_{m \in \mathcal{M}_i} \sum_{n \in \mathcal{M}_j} x_{imjnv} + \sum_{v \in \mathcal{V}} \sum_{m \in \mathcal{M}_i} \sum_{n \in \mathcal{M}_j} x_{jnimv} \leq 1 \quad i, j \in \mathcal{N} \; \Big| J_i = J_j \tag{22}$$

Constraints (21) force the x_{imjnv}-variables to take values closer to one or zero in the linear relaxation. For instance, for a station to be perfectly rebalanced, the sum over the x_{imjnv}-variables associated with that station must equal one. In [6] it is shown that the arcs between two stations of similar type need not be traversed more than once, resulting in constraints (22).

Table 1 includes a comparison of this mathematical model with the models in previous studies.

4 Computational Study

The mathematical model presented in Section 3 has been implemented in Xpress-IVE 1.24.06 using the Mosel programming language. The computational experiments have been executed on a computer with Intel Core i7-3770 3.40 GHz processor, 16 GB of RAM and running Windows 7.

4.1 Test Instances

Based on the BSS in Oslo, Norway, six test areas (geographical regions) have been identified. Details about the areas can be found in Table 4. The areas have an estimated optimal state for each station and a driving time matrix, T_{ij}^{D}. A parking time, T^{P}, set to one minute, is added for each station, while the handling time for each bike, T^{H}, is set to 30 seconds. All areas have two service vehicles. For each area, three instances are created by varying the initial states, while all other parameters are unchanged. Note that we assume perfect re-balancing for the third instance in each area, making the instances easier to solve because of a simpler structure.

Table 4. Test areas

Area	$\lvert\mathcal{N}\rvert$	**Avg. driving time**	$\overline{T}$	$\lvert\mathcal{V}\rvert$	**Cap.** $v=1$	**Cap.** $v=2$
1	6	2 min	16 min	2	10	10
2	8	6 min	30 min	2	10	15
3	10	6 min	40 min	2	12	12
4	12	5 min	30 min	2	10	10
5	14	7 min	45 min	2	12	12

4.2 Computational Results

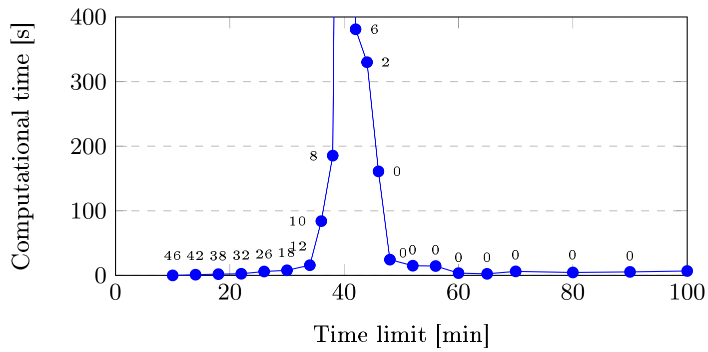
Various parameters in the model affect the computational time; the time limit, $\overline{T}$, the number of stations, $|\mathcal{N}|$, the maximum possible number of visits to each station, $|\mathcal{M}_i|$, and the number of service vehicles, $|\mathcal{V}|$. Among these, the time limit and the maximum possible number of visits are studied here.

Figure 3 shows that the computational time peaks when the time limit is set so that the total deviation is slightly above zero. By only changing the time limit, the computational time varies from less than one second to more than 35 minutes. The same pattern is seen for all instances.

The use of station visit numbers, $m, n \in \mathcal{M}_i$, is a new approach for the SBRP, allowing multiple station visits without a time-index. Though this formulation has some advantages, both the solution and the computational time depends on the value of $|\mathcal{M}_i|$, i.e. the maximum possible number of visits to each station. Each possible station visit (i,m) could be considered a distinct node in the graph. Hence, adding one element to the set $\mathcal{M}_i$ for one station i, is equivalent to adding a node to the graph.

Consequences of using different values for $|\mathcal{M}_i|$ is illustrated in Table 5. The *lower bound method* is the smallest number of visits to each station to allow perfect re-balancing, defined as: $|\mathcal{M}_i| = \left\lceil \frac{|I_i - O_i|}{\min_{v\in\mathcal{V}} C_v^{\mathcal{V}}} \right\rceil$. The *lower bound +1*

Fig. 3. The computational time depicted for different time limits, $\overline{T}$, for instance 4.1, i.e. the first instance from area 4. The numbers beside the markers indicate the total deviation between the initial and optimal state in the solution.



method allows one more visit to each station than the *lower bound method*. The *upper bound method* is defined as $|\mathcal{M}_i| = |I_i - O_i|$. For all our test instances the total number of deviations at the stations were the same for every method, independent of $|\mathcal{M}_i|$, hence only improvement in driving time is recorded in the table. Consequently, the *lower bound method* is recommended as it yields near optimal solutions with much less computational effort.

Depending on the input parameters, the mathematical model from Section 3 can be solved to optimality for instances of about 15 stations. Combined with some form of clustering, this could be enough to solve many realistically sized instances.

4.3 Comparison with Rules of Thumb

Today, in the Oslo BSS, the operators utilize their experience and common sense to decide the routes and the pickup and delivery quantities. Here, two greedy rules of thumb are created to imitate the operators behavior. The first rule of thumb states that the service vehicle should visit the nearest pickup and delivery stations in sequence, unless it is able to meet the demand at two subsequent stations of the same type. The vehicle should serve the entire demand of bikes at the stations, but is restricted by its capacity and the time limit for re-balancing. The second rule of thumb works quite similar, but the vehicle always goes to the station with the largest deviation.

A comparison is made between the results obtained with these rules of thumb and the ones obtained by solving the model from Section 3. The comparison is only made for instances 2.1 and 3.1, and to simplify only one vehicle is used. With regard to deviations, the SBRP-model finds solutions that are between 20.0 and 56.6% better than the two rules of thumb. A characteristic for the optimal solution is that it has less slack in the time restriction than the rules of thumb.

Table 5. Comparison of number of nodes in the graph, computational times, and quality of solution for three different methods for setting the maximum possible number of visits, $\mathcal{M}_i$. The improvement in solution is relative to the *lower bound method*. Note that the deviation between the initial and optimal state is equal for all methods, hence improvement in solution only refers to driving time.

	Lower bound		Lower bound +1			Upper bound		
Instance	$\sum_{i\in\mathcal{N}} \lvert\mathcal{M}_i\rvert$	**Comp. time**	$\sum_{i\in\mathcal{N}} \lvert\mathcal{M}_i\rvert$	**Comp. time**	**Imprv. in sol.**	$\sum_{i\in\mathcal{N}} \lvert\mathcal{M}_i\rvert$	**Comp. time**	**Imprv. in sol.**
1.1	6	0.19s	12	5.60s	0.0 %	24	139.70s	0.0 %
1.2	8	0.34s	14	1.91s	0.0 %	38	47.40s	0.0 %
1.3	8	0.20s	14	0.23s	0.0 %	48	>3000s	≥0.0 %
2.1	8	0.64s	16	697.00s	0.0 %	32	>3000s	≥0.0 %
2.2	9	0.44s	17	1.51s	0.0 %	46	>3000s	≥0.0 %
2.3	10	0.62s	18	7.81s	3.6 %	48	462.00s	3.6 %
3.1	10	1.25s	20	279.50s	0.0 %	56	>3000s	≥0.0 %
3.2	12	7.00s	22	281.00s	0.0 %	64	>3000s	≥0.0 %
3.3	12	1.25s	22	74.70s	0.0 %	58	>3000s	≥0.0 %
4.1	12	8.40s	24	>3000s	≥0.0 %	52	>3000s	≥0.0 %
4.2	15	17.00s	27	2089.00s	0.0 %	74	>3000s	≥0.0 %
4.3	12	0.40s	24	20.30s	3.3 %	62	286.50s	3.3 %
5.1	14	69.00s	28	>3000s	≥0.0 %	70	>3000s	≥0.0 %
5.2	16	15.30s	30	2708.00s	0.0 %	86	>3000s	≥0.0 %
5.3	16	1.07s	30	57.70s	7.7 %	106	>3000s	≥7.7 %
Average	n/a	8.21s	n/a	>814.95 s	≥1.6 %	n/a	>2262.00 s	≥1.6 %

4.4 Practical Use of the Model

Six of the 13 articles listed in Table 1 minimize time usage or cost, given that the system will be perfectly re-balanced. By assuming zero deviation, several simplifications can be made, and the computational time will decrease significantly, as indicated in Figure 3.

It is possible to utilize intervals, rather than a fixed number, to describe the optimal state. This provides more flexibility to the model, presumably making it harder to solve, but it may be more realistic. An alternative to use intervals, is to punish large deviations relatively more than small, for example by punishing the square of the deviation.

In addition to serving as a tool for operational planning, the SBRP-model could be used to support both strategic and tactical decisions. Analyzing changes in parameter values can be done by re-solving the problem for different values. By increasing the time limit for re-balancing operations, the number of deviations could go down. The operator may use this information to decide whether to expand the time limit or not. To support the decision of whether to acquire or dispose a service vehicle, the SBRP-model may be used to quantify the effect.

Increased vehicle capacity leads, as expected, to a reduced objective value. At a certain point, the objective value reaches its lowest point, where the total deviation is zero or the time limit restricts the objective value from decreasing further. To compare a change in the objective value with the cost of changing a parameter, the system operator is referred to a cost–benefit analysis.

5 Concluding Remarks

As the SBRP is a relatively novel problem, a review of the research made on the topic is missing in the literature. An extensive literature survey, consisting of the review and systematic comparison of 13 studies, has therefore been conducted. As can be seen from Table 1, many studies make assumptions that are unrealistic for most practical problems. We have proposed a new mathematical model for the SBRP that makes fewer assumptions and allows more possibilities than many existing models. For instance, this model allow a heterogeneous fleet, multiple visits to each station, and non-perfect re-balancing.

Since we have focused on the modeling and not on solution algorithms in this study, we are only able to solve relatively small instances. The model should however provide a good starting point for proposing more advanced solution methods, for instance as an important part of a clustering algorithm for solving realistically sized instances.

References

1. Angeloudis, P., Hu, J., Bell, M.G.: A strategic repositioning algorithm for bicycle-sharing schemes. Transportmetrica A: Transport Science 10(8), 759–774 (2014)
2. Benchimol, M., Benchimol, P., Chappert, B., De La Taille, A., Laroche, F., Meunier, F., Robinet, L.: Balancing the stations of a self service “bike hire” system. RAIRO-Operations Research 45(1), 37–61 (2011)
3. Berbeglia, G., Cordeau, J.F., Gribkovskaia, I., Laporte, G.: Static pickup and delivery problems: a classification scheme and survey. Top 15(1), 1–31 (2007)
4. Brinkmann, J., Ulmer, M.W., Mattfeld, D.C.: Short-term strategies for stochastic inventory routing in bike sharing systems. Transportation Research Procedia 10, 364–373 (2015)
5. Caggiani, L., Ottomanelli, M.: A modular soft computing based method for vehicles repositioning in bike-sharing systems. Procedia-Social and Behavioral Sciences 54, 675–684 (2012)
6. Chemla, D., Meunier, F., Calvo, R.W.: Bike sharing systems: Solving the static rebalancing problem. Discrete Optimization 10(2), 120–146 (2013)
7. Contardo, C., Morency, C., Rousseau, L.M.: Balancing a dynamic public bike-sharing system. Tech. Rep. CIRRELT-2012-09, Universitè de Montrèal, Montrèal, Canada (2012), http://claudio.contardo.org/wp-content/uploads/2011/09/CIRRELT-2012-09.pdf
8. Dell'Amico, M., Hadjicostantinou, E., Iori, M., Novellani, S.: The bike sharing rebalancing problem: Mathematical formulations and benchmark instances. Omega 45, 7–19 (2014)

9. DeMaio, P.: Bike-sharing: History, impacts, models of provision, and future. Journal of Public Transportation 12(4), 3 (2009)
10. DeMaio, P., Meddin, R.: The bike-sharing world map. Available at: www.bikesharingmap.com (2015), (Accessed: 2015-October-06)
11. Desrochers, M., Laporte, G.: Improvements and extensions to the Miller-Tucker-Zemlin subtour elimination constraints. Operations Research Letters 10(1), 27–36 (1991)
12. Erdoğan, G., Battarra, M., Calvo, R.W.: An exact algorithm for the static rebalancing problem arising in bicycle sharing systems. European Journal of Operational Research 245(3), 667–679 (2015)
13. Erdoğan, G., Laporte, G., Calvo, R.W.: The static bicycle relocation problem with demand intervals. European Journal of Operational Research 238(2), 451–457 (2014)
14. Forma, I.A., Raviv, T., Tzur, M.: A 3-step math heuristic for the static repositioning problem in bike-sharing systems. Transportation Research Part B: Methodological 71, 230–247 (2015)
15. Fricker, C., Gast, N.: Incentives and redistribution in homogeneous bike-sharing systems with stations of finite capacity. EURO Journal on Transportation and Logistics 3, 1–31 (2014)
16. García-Palomares, J.C., Gutiérrez, J., Latorre, M.: Optimizing the location of stations in bike-sharing programs: a GIS approach. Applied Geography 35(1), 235–246 (2012)
17. Gaspero, L., Rendl, A., Urli, T.: Balancing bike sharing systems with constraint programming. Constraints 20, 1–31 (2015)
18. Hernández-Pérez, H., Salazar-González, J.J.: The one-commodity pickup-and-delivery traveling salesman problem: Inequalities and algorithms. Networks 50(4), 258–272 (2007)
19. Ho, S.C., Szeto, W.: Solving a static repositioning problem in bike-sharing systems using iterated tabu search. Transportation Research Part E: Logistics and Transportation Review 69, 180–198 (2014)
20. Kaspi, M., Raviv, T., Tzur, M.: Detection of unusable bicycles in bike-sharing systems. Available at: http://www.eng.tau.ac.il/ talraviv/Publications/Detection%20of%20Unusable%20Bicycles.pdf (2015), working paper. Tel-Aviv University. (Published: 2015-October, Accessed: 2015-November-08)
21. Kloimüllner, C., Papazek, P., Hu, B., Raidl, G.R.: Balancing bicycle sharing systems: an approach for the dynamic case. In: Evolutionary Computation in Combinatorial Optimisation, pp. 73–84. Springer (2014)
22. Laporte, G., Meunier, F., Wolfler Calvo, R.: Shared mobility systems. 4OR 13(4), 341–360 (2015)
23. Lin, J.R., Yang, T.H.: Strategic design of public bicycle sharing systems with service level constraints. Transportation Research Part E: Logistics and Transportation Review 47(2), 284–294 (2011)
24. Midgley, P.: Bicycle-sharing schemes: enhancing sustainable mobility in urban areas. In: 19th session of the Commission on Sustainable Development. 2011-May-02. United Nations, Department of Economic and Social Affairs (05 2011), background Paper No. 8
25. Miller, C.E., Tucker, A.W., Zemlin, R.A.: Integer programming formulation of traveling salesman problems. Journal of the ACM (JACM) 7(4), 326–329 (1960)

26. Nair, R., Miller-Hooks, E., Hampshire, R.C., Bušić, A.: Large-scale vehicle sharing systems: analysis of Vélib'. International Journal of Sustainable Transportation 7(1), 85–106 (2013)
27. O'Mahony, E., Shmoys, D.B.: Data analysis and optimization for (citi) bike sharing. In: Twenty-Ninth AAAI Conference on Artificial Intelligence. 2015-January-25. Association for the Advancement of Artificial Intelligence (01 2015)
28. Rainer-Harbach, M., Papazek, P., Hu, B., Raidl, G.: Balancing bicycle sharing systems: A variable neighborhood search approach. In: Middendorf, M., Blum, C. (eds.) Evolutionary Computation in Combinatorial Optimization, Lecture Notes in Computer Science, vol. 7832, pp. 121–132. Springer Berlin Heidelberg (2013)
29. Rainer-Harbach, M., Papazek, P., Raidl, G.R., Hu, B., Kloimüllner, C.: PILOT, GRASP, and VNS approaches for the static balancing of bicycle sharing systems. Journal of Global Optimization 63(3), 1–33 (2013)
30. Raviv, T., Kolka, O.: Optimal inventory management of a bike-sharing station. IIE Transactions 45(10), 1077–1093 (2013)
31. Raviv, T., Tzur, M., Forma, I.A.: Static repositioning in a bike-sharing system: models and solution approaches. EURO Journal on Transportation and Logistics 2(3), 187–229 (2013)
32. Regue, R., Recker, W.: Proactive vehicle routing with inferred demand to solve the bikesharing rebalancing problem. Transportation Research Part E: Logistics and Transportation Review 72, 192–209 (2014)
33. Romero, J.P., Ibeas, A., Moura, J.L., Benavente, J., Alonso, B.: A simulation-optimization approach to design efficient systems of bike-sharing. Procedia-Social and Behavioral Sciences 54, 646–655 (2012)
34. Schuijbroek, J., Hampshire, R., van Hoeve, W.J.: Inventory rebalancing and vehicle routing in bike sharing systems. Available at: http://repository.cmu.edu/cgi/viewcontent.cgi?article=2490&context=tepper (2013), working paper. Tepper School of Business. (Published: 2013-February, Accessed: 2015-December-01)
35. Shaheen, S., Guzman, S., Zhang, H.: Bikesharing in Europe, the Americas, and Asia: past, present, and future. Transportation Research Record: Journal of the Transportation Research Board (2143), 159–167 (2010)
36. Sörensen, K., Dilip, D.: The (city) bike request scheduling problem–a novel approach to solve the city bike repositioning problem. In: Toklu, Y.C., Bekdas, G. (eds.) Metaheuristics and Engineering, Workshop of the EURO Working Group, vol. 15, pp. 157–161. Bilecik Şeyh Edebali University (2014)
37. Vogel, P., Ehmke, J.F., Mattfeld, D.C.: Service network design of bike sharing systems. Available at: https://www.tu-braunschweig.de/Medien-DB/winfo/ publications/service_network_design_of_bike_sharing_systems.pdf (2015), working paper. Technische Unversität Braunschweig. (Published: 2015-March-24, Accessed: 2015-September-25)
38. Vogel, P., Greiser, T., Mattfeld, D.C.: Understanding bike-sharing systems using data mining: Exploring activity patterns. Procedia-Social and Behavioral Sciences 20, 514–523 (2011)
39. Vogel, P., Neumann Saavedra, B., Mattfeld, D.: A hybrid metaheuristic to solve the resource allocation problem in bike sharing systems. In: Blesa, M., Blum, C., Voß, S. (eds.) Hybrid Metaheuristics, Lecture Notes in Computer Science, vol. 8457, pp. 16–29. Springer International Publishing (2014)