

A robust MPC approach for the rebalancing of mobility on demand systems

Original

A robust MPC approach for the rebalancing of mobility on demand systems / Calafiore, G.C., Bongiorno, C., Rizzo, A.. - In: CONTROL ENGINEERING PRACTICE. - ISSN 0967-0661. - ELETTRONICO. - 90:September 2019(2019), pp. 169-181. [10.1016/j.conengprac.2019.06.015]

Availability:

This version is available at: 11583/2742689 since: 2019-09-17T16:12:09Z

Publisher:

Elsevier

Published

DOI:10.1016/j.conengprac.2019.06.015

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

Elsevier postprint/Author's Accepted Manuscript

© 2019. This manuscript version is made available under the CC-BY-NC-ND 4.0 license
<http://creativecommons.org/licenses/by-nc-nd/4.0/>. The final authenticated version is available online at:
<http://dx.doi.org/10.1016/j.conengprac.2019.06.015>

(Article begins on next page)

A Robust MPC Approach for the Rebalancing of Mobility on Demand Systems

Giuseppe C. Calafiore^{a,b}, Christian Bongiorno^a, Alessandro Rizzo^{a,*}

^a*Dipartimento di Elettronica e Telecomunicazioni - Politecnico di Torino - Torino - Italy*

^b*IEIT-CNR Torino, Italy*

Abstract

A control-oriented model for mobility-on-demand systems is here proposed. The system is first described through dynamical stochastic state-space equations, and then suitably simplified in order to obtain a control-oriented model, on which two control strategies based on Model Predictive Control are designed. The former aims at keeping the expected value of the number vehicles parked in stations within prescribed bounds; the latter specifically accounts for stochastic fluctuations about the expected value. The model includes the possibility of weighting the control effort, leading to control solutions that may trade off efficiency and cost. The models and control strategies are validated over a dataset of logged trips of ToBike, the bike-sharing systems in the city of Turin, Italy.

Keywords:

Mobility on Demand, Bike-sharing, Rebalancing, Traffic Models, Nonlinear Models, Data Analysis, Model Predictive Control

*Corresponding Author

Email address: alessandro.rizzo@polito.it (Alessandro Rizzo)

URL: <http://staff.polito.it/alessandro.rizzo> (Alessandro Rizzo)

1. Introduction

Mobility-on-Demand (MOD) systems are becoming pervasive in cities of any size. As of December 2016, bike- and car-sharing programs had been adopted by more than 1,000 cities worldwide [1]. As of July 2017, Car2Go, the largest car-sharing company in the world, has 2,500,000 registered users and a fleet of nearly 14,000 vehicles in 26 locations in North America, Europe, and Asia [2].

The concept behind a MOD system is straightforward: a user requires a vehicle, picks it up from a designated location, executes the trip, drops off the vehicle at her/his destination. MOD systems can be station-based, with vehicles parked at fixed locations (stations), or floating, with vehicles parked with no constraints, at the user's wish.

As required to all service providers, MOD systems should be designed to meet the customer demand, which is extremely heterogeneous due to several factors, such as the time of the day, the season, commuting patterns, up-hill or down-hill stations (for bikes) [3, 4, 5]. The impossibility of meeting the customer demand is usually caused by a lack of vehicles at some locations and a corresponding surplus of vehicles somewhere else. This issue can be mitigated through the implementation of repositioning policies, also called *rebalancing*. Rebalancing strategies are typically obtained as the solution of suitable optimization problems. In bike-sharing systems, most of the works in the literature assume that one or more trucks are available to redistribute bikes over the city, aiming at maximizing the system performance while keeping the repositioning effort at a minimum. In car-sharing systems, rebalancing is operated on a single-vehicle basis by operators [6].

Rebalancing is often executed during time periods where traffic is low, especially at night. This activity, called *static rebalancing*, assumes that vehicles are

26 not used by customers during repositioning operations, or that their use is negli-
27 ble with respect to the rebalancing flows. In bike-sharing systems, repositioning
28 is usually executed by trucks able to displace high volumes of bikes, even within
29 relatively long distances [7, 8, 9, 10, 11, 12, 12, 13, 14, 15, 16]. *Dynamic repositi-*
30 *oning*, on the other hand, assumes that customers are traveling while rebalancing
31 operations occur, and the effects of such travels are not negligible. This kind of
32 repositioning is usually performed with smaller vehicles and/or over shorter dis-
33 tances [17, 18, 19, 20]. Users may also be involved in system rebalancing through
34 incentives [21]. For example, in Paris, Velib+ offers rewards to people moving
35 bikes up-hill [22].

36 Optimization algorithms informing repositioning strategies are based on suit-
37 able models of the MOD system. Several modeling techniques have been pro-
38 posed in the literature, mostly based on statistical and data-driven approaches, to
39 account for the stochasticity of the system under exam [23, 24, 25, 26, 27]. Most
40 of these works, however, rely on models with limited analytical tractability, due
41 to the lack of specific dynamical equations.

42 In this paper, a novel control-oriented model for a station-based MOD sys-
43 tem is proposed, extending preliminary results presented in [28]. In its general
44 formulation, the system is described by a dynamical model with stochastic state
45 variables in discrete time. The model quantifies vehicle flows between different
46 locations and accounts for stochasticity in customer demand and traveling times.
47 Hence, it is used to derive two MPC-based techniques [29]: first, a linear and time-
48 invariant approximation of the model is derived. Such a model is used to control
49 the expected values of the state variables, representing the expected quantity of
50 vehicles at each station. Then, interval analysis is leveraged to develop a robust

51 control strategy on the stochastic model, thus accounting for the fluctuations about
52 the expected values.

53 The design of the mentioned control strategies yields constrained optimiza-
54 tion problems, with the objective of maintaining the number of vehicles within
55 prescribed bounds at each location. As a solution to these problems, vehicle flows
56 that tend to rebalance the system are derived. These flows can then be used to
57 devise incentive-based strategies, or to set up rebalancing campaigns operated by
58 trucks or other transportation means.

59 Even though the proposed approach is conceived for station-based systems, it
60 can be effectively applied to free-floating systems, via a suitable tessellation of the
61 operational space. Location-based rebalancing can be also useful when stations
62 are close to each other, and users can effortlessly move through groups of stations
63 to seek for suitable pick-up or drop-off locations. This is the case, for example, of
64 the ToBike bike-sharing systems, located in the city of Turin (Italy), on which the
65 proposed approach is validated using one year of logged user trip data.

66 This paper extends preliminary results presented in [28] along several direc-
67 tions: *i*) the dynamical model is better formalized, and a more in-depth theoretical
68 treatment is offered; *ii*) the control of the expected value is expanded and includes
69 now a detailed study of steady-state conditions; *iii*) a robust control strategy based
70 on interval modeling is devised; *iv*) model and control validation are performed
71 on a real dataset of one year of logged trips provided by ToBike.

72 The paper is structured as follows. In Section 2 the modeling rationale, model
73 parameters, nomenclature, and modeling assumptions toward the realization of a
74 control-oriented model are given. Section 3 introduces the control-oriented model,
75 studying in detail the dynamics of the expected value of the state variables and the

76 steady-state conditions. In Section 4, two MPC-based control strategies, first on
 77 the expected value of the space variables, and then, on robust bounds of the state
 78 variables, are proposed. The ToBike MOD system and the dataset adopted for val-
 79 idation, along with relevant data analysis techniques are described in Section 5.
 80 Section 6 comprehensively validates our model and assesses its performance, us-
 81 ing the ToBike dataset. Conclusions are finally drawn in Section 7.

82 **2. Control- and simulation-oriented models of the MOD system**

83 A MOD system is here modeled as a network composed by nodes, represent-
 84 ing the vehicle stations, and links between nodes, representing the vehicle routes.
 85 The set \mathcal{S} of station nodes is composed by N stations, and the set \mathcal{L} is composed
 86 by N^2 links between any two station nodes in \mathcal{S} . A link (i, j) between departure
 87 station i and destination station j does not necessarily represent a specific physical
 88 route, but rather the ensemble of all routes that are typically traveled by customers
 89 moving from i to j .

90 The generic link (i, j) is characterized by its state $v_{ij}(t)$, representing the car-
 91 dinality of the set $\mathcal{V}_{ij}(t)$ of all the vehicles *en route* from i to j at time t , and by the
 92 (random) fraction $\tilde{q}_{ij}(t, \delta) \in [0, 1]$ of the $v_{ij}(t)$ vehicles that reach their destination
 93 j within the time interval $(t, t + \delta]$.

94 The generic node i is characterized by its state $z_i(t)$, denoting the number of
 95 vehicles parked (and hence available for pick up) at station i at time t , and by
 96 the instantaneous mean rate $\mu_i(t) \in \mathbb{R}^+$ of random service requests that arrive at
 97 station i at time t . Analogously, $1/\mu_i(t)$ describes the mean of the random inter-
 98 arrival time of service requests at station i . The station throughput $\lambda_i(t)$ is instead
 99 the mean rate at which vehicles depart from station i . In reality, it always holds

100 that $\lambda_i(t) \leq \mu_i(t)$, since not all the service requests may be fulfilled, due to the
 101 fact that there may exist station-empty periods in which no vehicles are available
 102 at the station, and hence no departure is possible from the station, even if demand
 103 from customers exists. This issue is further discussed in Section 2.1.4.

104 If a customer request is generated at station i at time t , and if station i has a
 105 vehicle available, then the customer picks up that vehicle and starts a trip towards
 106 a destination station j . The selection of the destination is modeled via a set of
 107 (possibly time-varying) *routing probabilities*, i.e., it is assumed that at time t and
 108 for each station i , there exist probabilities $p_{ij}(t) \in \mathbb{R}^+$, with $\sum_j p_{ij}(t) = 1$, such
 109 that a generic customer departing from i chooses destination j with probability
 110 $p_{ij}(t)$.

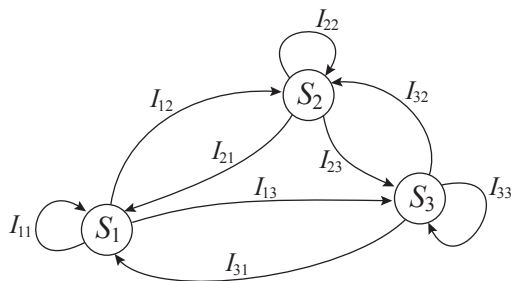


Figure 1: Example of network modeling a MOD system with three stations S_1, S_2, S_3 , and corresponding itinerary links I_{ij} .

111 *2.1. Simplifying assumptions for a “control-oriented” model*

112 A real-life MOD system is hardly representable by a simple mathematical
 113 model. For one thing, any stochastic representation of, say, the transit times or
 114 of the customer requests, is only an approximation of reality. Also, a key aspect
 115 in station-based MOD systems is that a station state $z_i(t)$ may reach its physical

116 limits, that is, $z_i(t)$ may be equal to zero (empty state) or to c_i^{\max} (full state), where
117 c_i^{\max} is the maximum capacity of the station (typically, the number of physical
118 slots available for parking the vehicles in station i). When $z_i(t) = 0$, customer
119 requests at the i -th station cannot be fulfilled and must be turned down. Similarly,
120 when $z_i(t) = c_i^{\max}$, vehicles arriving at station i at time t are unable to park their
121 vehicles, and must therefore wait, or be diverted to some other stations that have
122 available parking slots. The two issues just mentioned are indeed a manifestation
123 of a common problem in MOD systems known as “station imbalance,” a situation
124 in which certain stations are over-requested as departure stations, and thus quickly
125 become empty, and other stations are over-requested as destination stations, and
126 thus quickly become full. Strong imbalance severely affects the usability (and the
127 economic profitability) of the system.

128 Clearly, it is possible to build a sophisticated model that tries to mimic as
129 close as possible the behavior of the real system. This is what is usually called a
130 *simulation model*, which is needed for testing policies and assessing the system
131 performance via simulations. Such a model is a proxy of reality, and it may be
132 used in place of reality, whenever this is needed. The model presented in this
133 paper, on the other hand, has a different goal: to develop a *control model*, that
134 is, a model whose primary purpose is to help in the design of control strategies
135 (e.g., rebalancing policies) for the system. A good control model should be simple
136 enough to allow for the effective synthesis of the control law, although this may
137 come at the expense of approximations. In the end, however, the performance of
138 the control law should be tested and evaluated on the real system, or on a proxy
139 of it (i.e., the simulation model).

140 With this in mind, the main simplifying assumptions used to construct the

141 control-oriented model of the MOD system are introduced in the following.

142 2.1.1. Piece-wise constant parameters

143 The model discussed so far has time-varying parameters, matching the fact
144 that rates, routing probabilities and transit times of a real-world system are likely
145 not constant over time. However, dealing with continuously time-varying parame-
146 ters is impractical for our purposes of system optimization. Previous analyses per-
147 formed on logged service data suggest that system parameters can be conveniently
148 approximated as piece-wise constant functions [23, 24], where each constant part
149 corresponds to a subset of the day. Motivated by this observation, constant param-
150 eters over each given period are here considered.

151 2.1.2. Exponential inter-departure times

152 Departures from each station i follow a counting process with instantaneous
153 rate $\lambda_i(t)$. For simplicity, it is specifically assumed that they form a Poisson
154 process of rate $\lambda_i(t)$, although this latter assumption is not critical for our de-
155 velopments. Assuming that each vehicle departing from i at time t chooses its
156 destination j with probability $p_{ij}(t)$ implies that vehicles departing from i with
157 destination j at time t also follow a Poisson process with rate $p_{ij}(t)\lambda_i(t)$.

158 2.1.3. Densities for link arrival proportions

As previously discussed, the transit of vehicles through the (i, j) link is mod-
eled by assuming that, at each given t and given δ , only a (random) fraction
 $\tilde{q}_{ij}(t, \delta) \in [0, 1]$ of the $v_{ij}(t)$ vehicles reach their destination j within the time
interval $(t, t + \delta]$. Letting

$$q_{ij}(t, \delta) \doteq \mathbb{E} \{ \tilde{q}_{ij}(t, \delta) \} \in [0, 1],$$

159 it is assumed that $\tilde{q}_{ij}(t, \delta)$ is statistically independent from $v_{ij}(t)$. The characteri-
 160 zation of $\tilde{q}_{ij}(t, \delta)$ should be done via a specific statistical analysis of log data from
 161 the real (or simulated) system. If the average transit times $\tau_{ij}(t)$ are estimated for
 162 each link (i, j) , for instance, then one may choose the expected proportion as
 163 $q_{ij}(t, \delta) = 1 - e^{-\delta/\tau_{ij}(t)}$, or simply as $q_{ij}(t, \delta) = \delta/\tau_{ij}(t)$, for $\delta \leq \tau_{ij}(t)$.

164 2.1.4. No blocking

165 It is assumed, for the purpose of the control model, that stations have unlimited
 166 capacity, i.e., $c_i^{\max} = +\infty, \forall i$. Also, it is fictiously assumed that customer demand
 167 is always satisfied. The motivation for these assumptions is explained in the fol-
 168 lowing. In reality, a station state $z_i(t)$ remains bounded in $[0, c_i^{\max}]$ at all times.
 169 When one of the two limits in the interval is attained, the station is “blocked”, in
 170 the sense that no operations can occur until an event (i.e., a departure for a full
 171 station or arrival for an empty station) that takes the station state back within the
 172 prescribed capacity limits happens. Reproducing the “blocking” condition in our
 173 control-oriented model would prevent us from developing a time-driven model
 174 and hinder the tractability of the model. Hence, we here admit that variables
 175 $z_i(t)$ are allowed to go beyond the boundaries, while penalizing out-of-boundary
 176 behavior is penalized in the control design phase. Thus, the control action is rein-
 177 forced until the imbalance situation is compensated. Letting $z_i(t) > c_i^{\max}$ implies,
 178 in practice, to assume that there are always enough parking slots at the station,
 179 and letting $z_i(t) < 0$ implies to assume that there are always enough vehicles at
 180 the stations; this just implies that unavailable vehicles are borrowed from some-
 181 where in order to satisfy the demand. The rebalancing control policy, however,
 182 will be designed to minimize such out-of-boundary situations.

183 **3. The MOD control model**

184 Under the assumptions of Section 2.1, the following quantities are defined:

- 185 • Δ is the time period during which the system parameters are assumed to be
186 constant, as discussed in Section 2.1.1.
- 187 • n_h is a convenient positive integer number used to divide the whole simula-
188 tion time T in equally spaced time intervals of duration $\delta = T/n_h$.
- $d_{ij}(t + \delta)$ is the number of vehicles driven by users that depart from i with destination j in the time interval $(t, t + \delta]$. According to the assumption in Section 2.1.2, $d_{ij}(t + \delta)$ has a Poisson probability mass

$$\text{Prob}\{d_{ij}(t + \delta) = k\} = \frac{1}{k!} (p_{ij}(t)\lambda_i(t)\delta)^k e^{-p_{ij}(t)\lambda_i(t)\delta},$$

for $k = 0, 1, \dots$ (1)

- 189 • $r_{ij}(t)$ is the number of “control” vehicles (i.e., vehicles used for rebalancing
190 purposes) that are moved from i to j as dictated by the rebalancing control
191 strategy. From a modeling point of view, these are considered to be “in-
192 jected” in the (i, j) link during the interval $(t, t + \delta]$. Contrary to $d_{ij}(t + \delta)$,
193 $r_{ij}(t)$ is a deterministic quantity that is determined by the control algorithm
194 at time t . It is assumed that δ is sufficiently small with respect to the aver-
195 age link transit times τ_{ij} , so that there is a (practically) zero probability that
196 any of the $d_{ij}(t + \delta)$ or of the $r_{ij}(t)$ vehicles reaches its destination by time
197 $t + \delta$.
- $a_{ij}(t + \delta)$ is the number of vehicles, among the ones in $\mathcal{V}_{ij}(t)$, that reach the j -th station by time $t + \delta$. According to the assumption in Section 2.1.3, the

count $a_{ij}(t + \delta)$ can be written equivalently in terms of the *random proportion* $\tilde{q}_{ij}(t, \delta)$ as

$$a_{ij}(t + \delta) = \tilde{q}_{ij}(t, \delta)v_{ij}(t), \quad (2)$$

198 where $\tilde{q}_{ij}(t, \delta)$ is a random parameter with expected value $q_{ij}(t, \delta)$. Equa-
 199 tion (2) simply states that the number of vehicles in $\mathcal{V}_{ij}(t)$ that reach the j -th
 200 station by time $t + \delta$ is a (random) fraction of the whole number of vehicles
 201 in $\mathcal{V}_{ij}(t)$, and this fraction is, on average, equal to $q_{ij}(t, \delta)$. We observe that
 202 the quantity $a_{ij}(t + \delta)$ is implicitly allowed to be real valued.

203 Table 1 summarizes the notation of main variables used in this paper.

204 The discrete-time equations that regulate the system behavior can now be writ-
 205 ten. For $i, j = 1, \dots, N$, straightforward conservation arguments yield

$$\begin{aligned} v_{ij}(t + \delta) &= v_{ij}(t) - a_{ij}(t + \delta) + d_{ij}(t + \delta) + r_{ij}(t) \\ z_j(t + \delta) &= z_j(t) + \sum_i a_{ij}(t + \delta) - \sum_h (d_{jh}(t + \delta) + r_{jh}(t)). \end{aligned}$$

206 The following equations are obtained using (2):

$$\begin{aligned} v_{ij}(t + \delta) &= (1 - \tilde{q}_{ij}(t, \delta))v_{ij}(t) + \\ &\quad + d_{ij}(t + \delta) + r_{ij}(t) \end{aligned} \quad (3)$$

$$\begin{aligned} z_j(t + \delta) &= z_j(t) + \sum_i \tilde{q}_{ij}(t, \delta)v_{ij}(t) + \\ &\quad - \sum_h (d_{jh}(t + \delta) + r_{jh}(t)). \end{aligned} \quad (4)$$

207 The system above is a linear, discrete-time, stochastic one in the z_j and v_{ij} state
 208 variables, with stochastic inputs given by the d_{ij} departures, and control inputs
 209 given by the rebalancing departures r_{ij} . Given initial conditions, Eqs. (3)-(4) can
 210 be used to propagate forward in time the (random) system states. In the following,

Variable	Description
N	Number of stations
t	Discrete time index
Δ	Time interval in which system parameters are assumed to be constant
δ	Sampling interval
n_h	Time horizon used for optimization in MPC
$v_{ij}(t)$	Number of vehicles <i>en route</i> from station i to station j at time t
$\tilde{q}_{ij}(t, \delta)$	Fraction of $v_{ij}(t)$ vehicles that reach station j within the interval $(t, t + \delta)$
$q_{ij}(t, \delta)$	Average value of $\tilde{q}_{ij}(t, \delta)$
$a_{ij}(t + \delta)$	Number of vehicles that reach station j from station i by time $t + \delta$
$\mu_i(t)$	Instantaneous mean rate of service request at station i at time t
$\lambda_i(t)$	Instantaneous mean rate of station i throughput
$p_{ij}(t)$	Routing probability from station i to station j at time t
c_i^{\min}, c_i^{\max}	Bounds for the desired number of vehicles available at station i
$r_{ij}(t)$	Number of “control” (rebalancing) vehicles to be displaced from station i to station j at time t

Table 1: Nomenclature of the main variables used in the paper.

211 the dynamics of the expected value of the model state variables will be derived and
212 used to design the first MPC-based controller.

213 4. MPC-based rebalancing

214 4.1. The expected state dynamics

215 Observing that $\sum_h d_{jh}(t + \delta)$ is Poisson with parameter $\sum_h p_{jh} \lambda_j \delta = \lambda_j \delta$,
216 denoting expected quantities with an overbar (i.e., $\bar{v}_{ij}(t) \doteq \mathbb{E}\{v_{ij}(t)\}$, $\bar{z}_j(t) \doteq$

217 $E\{z_j(t)\}$, etc.), and recalling that $\tilde{q}_{ij}(t, \delta)$ and $v_{ij}(t)$ are assumed to be statisti-
 218 cally independent, the evolution of the expected value of the state equations in
 219 (3)-(4) can be written as

$$\begin{aligned} \bar{v}_{ij}(t + \delta) &= (1 - q_{ij}(t, \delta))\bar{v}_{ij}(t) + \\ &\quad + p_{ij}(t)\lambda_i(t)\delta + r_{ij}(t) \end{aligned} \quad (5)$$

$$\begin{aligned} \bar{z}_j(t + \delta) &= \bar{z}_j(t) + \sum_i q_{ij}(t, \delta)\bar{v}_{ij}(t) + \\ &\quad - \lambda_j(t)\delta - \sum_h r_{jh}(t). \end{aligned} \quad (6)$$

220 Equations (5)-(6) constitute a linear, discrete-time, deterministic dynamical sys-
 221 tem in the expected state variables $\bar{z}_j(t)$ and $\bar{v}_{ij}(t)$, with inputs given by the mean
 222 departure rates $\lambda_i(t)$, and control inputs given by the rebalancing departures $r_{ij}(t)$.

223 4.2. Steady-state behavior

224 Assume that the rebalancing inputs are constant, i.e., $r_{ij}(t) = r_{ij}, \forall t$, and that
 225 the system parameters p_{ij}, q_{ij} , and $\lambda_j, i, j = 1, \dots, N$, remain constant. In such
 226 case, the following equations hold:

$$\bar{v}_{ij}(t + \delta) = (1 - q_{ij}(\delta))\bar{v}_{ij}(t) + p_{ij}\lambda_i\delta + r_{ij} \quad (7)$$

$$\bar{z}_j(t + \delta) = \bar{z}_j(t) + \sum_i q_{ij}(\delta)\bar{v}_{ij}(t) - \lambda_j\delta - \sum_h r_{jh}. \quad (8)$$

Since $1 - q_{ij}(\delta) < 1$, the discrete time recursion (7) is asymptotically stable, and
 has a constant input term equal to $p_{ij}\lambda_i\delta + r_{ij}$. Therefore, $\bar{v}_{ij}(t + k\delta)$ approaches
 a steady-state value for $k \rightarrow \infty$, independently from the initial condition $\bar{v}_{ij}(t)$, and
 this value is

$$\bar{v}_{ij}^{(ss)} = \frac{p_{ij}\lambda_i\delta + r_{ij}}{q_{ij}(\delta)}. \quad (9)$$

Plugging the steady-state value (9) into (8), a steady-state for $\bar{z}_j^{(ss)}$ exists when

$$\bar{z}_j^{(ss)} = \bar{z}_j^{(ss)} + \delta \sum_i p_{ij} \lambda_i - \lambda_j \delta + \sum_i r_{ij} - \sum_h r_{jh}.$$

227 This steady state condition is satisfied if the following flow equilibrium holds for
 228 all $j = 1, \dots, N$:

$$\delta \sum_i p_{ij} \lambda_i + \sum_i r_{ij} = \lambda_j + \sum_h r_{jh}. \quad (10)$$

229 Since (8) is a pure discrete-time integrator, the actual steady-state value depends
 230 on the initial conditions and on the transient of $\bar{v}_{ij}(t + k\delta)$. Incidentally, it can
 231 be observed that if $\delta \ll \Delta$, since the system parameters p_{ij} and λ_j , $i, j = 1, \dots, N$,
 232 remain constant within an interval of duration Δ , it may be assumed that $\bar{v}_{ij}(t + k\delta)$
 233 and $\bar{z}_j(t + k\delta)$ will rapidly attain their steady-state value.

234 4.3. Control of the expected value

The adopted cost function is the total weighted rebalancing effort over the considered time horizon

$$J_T = \sum_{k=0}^{n_h-1} \|\text{vec}(\mathbf{W} \circ \mathbf{R}(t + k\delta))\|_1,$$

where $\mathbf{R}(t + k\delta)$ is the matrix containing in row i and column j the rebalancing departures $r_{ij}(t + k\delta)$, $\mathbf{W} \in \mathbb{R}^{N \times N}$ is a weight matrix with nonnegative elements, the operator $\text{vec}(\cdot)$ indicates the operation of vectorization of a matrix, the operator “ \circ ” indicates the Hadamard product, and $\|\cdot\|_1$ indicates the 1-norm of a vector. The weight matrix can be used to specify the relative cost of rebalancing between different pairs of stations, due for example to high displacement time, level of traffic, steep routes, etc. For simplicity and to enhance readability, and without loss of

generality, in this work we set \mathbf{W} as the all-one matrix and we omit the vectorization notation, so that we define $\|\mathbf{R}(t+k\delta)\|_1 \doteq \|\text{vec}(\mathbf{R}(t+k\delta))\|_1$. The control goal is to maintain the expected states $\bar{z}_j(t+k\delta)$ within given limits $[c_j^{\min}, c_j^{\max}]$, at all times, while minimizing the rebalancing effort. Letting

$$\mathcal{R} \doteq \{\mathbf{R} \in \mathbb{R}^{N,N} : \mathbf{R} \geq 0, \text{ and } R_{ii} = 0, i = 1, \dots, N\},$$

235 the following optimization problem is solved:

$$\begin{aligned} \min_{\mathbf{R}(t), \dots, \mathbf{R}(t+(n_h-1)\delta) \in \mathcal{R}} \quad & \sum_{k=0}^{n_h-1} \|\mathbf{R}(t+k\delta)\|_1 \\ \text{s.t.} \quad & \bar{z}_j(t+k\delta) \in [c_j^{\min}, c_j^{\max}], \\ & \text{for } j = 1, \dots, N, \text{ and } k = 1, \dots, n_h, \end{aligned}$$

236 where $\bar{z}_j(t+k\delta)$ is given by the recursion in (5)-(6), initialized with given initial
237 conditions $\bar{z}_j(t), \bar{v}_{ij}(t), i, j = 1, \dots, N$.

Imposing strict feasibility for the state limits $\bar{z}_j(t+k\delta) \in [c_j^{\min}, c_j^{\max}]$ may result in infeasibility, or in a too high rebalancing effort for most of the possible scenarios. A more flexible approach is therefore to consider a trade-off between rebalancing effort and constraint satisfaction, by introducing slack variables $s_j(t+k\delta)$. Slack variables are used in this case to allow a certain degree of violation of the station capacity constraint against a mitigation of the control effort. The state constraint is firstly rewritten as

$$\left| \bar{z}_j(t+k\delta) - \frac{c_j^{\max} + c_j^{\min}}{2} \right| \leq \frac{c_j^{\max} - c_j^{\min}}{2},$$

and then the problem is relaxed to

$$\begin{aligned}
& \min \sum_{k=0}^{n_h-1} \sum_j s_j(t+k\delta) + \gamma \|\mathbf{R}(t+k\delta)\|_1 \\
& \text{s.t.: } \left| \bar{z}_j(t+k\delta) - \frac{c_j^{\max} + c_j^{\min}}{2} \right| \leq \frac{c_j^{\max} - c_j^{\min}}{2} + s_j(t+k\delta), \\
& s_j(t+\delta) \geq 0, \dots, s_j(t+n_h\delta) \geq 0, \\
& \text{for } j = 1, \dots, N, \text{ and } k = 1, \dots, n_h, \\
& \mathbf{R}(t), \dots, \mathbf{R}(t+(n_h-1)\delta) \in \mathcal{R},
\end{aligned} \tag{11}$$

238 where $\gamma \geq 0$ is a tunable trade-off parameter between the rebalancing effort and
239 satisfaction of the capacity constraints. Variables $s_j(t+k\delta)$ represent the amount
240 of allowed capacity violation at the j -th station at time $t+k\delta$. Ideally, the op-
241 timization problem should lead to a sharp satisfaction of the capacity constraint,
242 hence, the minimized cost function should entail slack variables equal to zero.
243 However, some scenarios might allow a much reduced control effort, if some of
244 the capacity constraints are relaxed. In this case, the optimization algorithm would
245 attain a value of the objective function such that some of the slack variables settle
246 to a nonzero value, implying that the capacity constraint is violated by an amount
247 equal to such a value. The trivial constraints on the nonnegativity of the slack
248 variables and on the feasibility of the control actions complete the definition of
249 the problem.

250 4.4. MPC based on robust interval modeling

251 Although the control procedure illustrated in Section 4.3 is an efficient way
252 to control the expected values of the state variables, the stochastic nature of the
253 system may lead to fluctuations that take the system dynamics out of the desired

254 behavior during specific realizations. This issue is here mitigated through the
 255 introduction of a robust control scheme based on interval analysis.

256 The stochastic information on $d_{ij}(t, \delta)$ in (1) can be used to obtain an interval
 257 of probabilistic confidence on this parameter, and hence it is possible to use inter-
 258 val analysis to unroll in time a confidence tube to bound the state of the dynamical
 259 system (3)-(4). Then, MPC will be applied to control the resulting dynamics. The
 260 obtained control scheme will guarantee the attainment of the control objective
 261 with the desired confidence level.

262 Fixing a probability level η , quantities $d_{ij}^{\min}(t, \delta)$ and $d_{ij}^{\max}(t, \delta)$ denote the
 263 extremes of confidence intervals of probability η for the random variable $d_{ij}(t, \delta)$.
 264 Due to the linearity of (3)-(4) in $d_{ij}(t, \delta)$, two conservative bounds within which
 265 state variables $v_{ij}(t)$ and $z_j(t)$ will be contained with probability η can be easily
 266 obtained. Denoting these bounds with $v_{ij}^{\min}(t)$, $v_{ij}^{\max}(t)$, $z_j^{\min}(t)$, and $z_j^{\max}(t)$,
 267 their dynamics is described by

$$\begin{aligned} v_{ij}^{\min}(t + \delta) &= (1 - \tilde{q}_{ij}(t, \delta))v_{ij}^{\min}(t) + \\ &\quad + d_{ij}^{\min}(t + \delta) + r_{ij}(t) \end{aligned} \quad (12)$$

$$\begin{aligned} z_j^{\min}(t + \delta) &= z_j^{\min}(t) + \sum_i \tilde{q}_{ij}(t, \delta)v_{ij}^{\min}(t) + \\ &\quad - \sum_h (d_{jh}^{\max}(t + \delta) + r_{jh}(t)) \end{aligned} \quad (13)$$

268 and

$$\begin{aligned} v_{ij}^{\max}(t + \delta) &= (1 - \tilde{q}_{ij}(t, \delta))v_{ij}^{\max}(t) + \\ &\quad + d_{ij}^{\max}(t + \delta) + r_{ij}(t) \end{aligned} \quad (14)$$

$$\begin{aligned} z_j^{\max}(t + \delta) &= z_j^{\max}(t) + \sum_i \tilde{q}_{ij}(t, \delta)v_{ij}^{\max}(t) + \\ &\quad - \sum_h (d_{jh}^{\min}(t + \delta) + r_{jh}(t)). \end{aligned} \quad (15)$$

269 The optimization problem that implements the MPC strategy is analogous to
 270 that of Section 4.3, with the difference that the algorithm considers now a larger
 271 set of constraints to impose that the two extremes of fluctuations are bounded
 272 between two given capacity limits:

$$\begin{aligned}
 & \min_{\mathbf{R}(t), \dots, \mathbf{R}(t+(n_h-1)\delta) \in \mathcal{R}} \quad \sum_{k=0}^{n_h-1} \|\mathbf{R}(t+k\delta)\|_1 \\
 & \text{s.t.} \quad z_j^{\min}(t+k\delta) \in [c_j^{\min}, c_j^{\max}], \\
 & \quad \quad z_j^{\max}(t+k\delta) \in [c_j^{\min}, c_j^{\max}], \\
 & \quad \quad \text{for } j = 1, \dots, N, \text{ and } k = 1, \dots, n_h.
 \end{aligned}$$

Similarly to Section 4.3, two sets of slack variables, denoted as $s_j^{\max}(t)$ and $s_j^{\min}(t)$, are used to establish a trade-off between feasibility, rebalancing effort, and constraint satisfaction. The optimization problem is therefore relaxed to

$$\begin{aligned}
 & \min \sum_{k=0}^{n_h-1} \sum_j [s_j^{\max}(t+k\delta) + s_j^{\min}(t+k\delta)] + \gamma \|\mathbf{R}(t+k\delta)\|_1 \\
 & \text{s.t.} \quad \left| z_j^{\max}(t+k\delta) - \frac{c_j^{\max} + c_j^{\min}}{2} \right| \leq \frac{c_j^{\max} - c_j^{\min}}{2} + s_j^{\max}(t+k\delta), \\
 & \quad \quad \left| z_j^{\min}(t+k\delta) - \frac{c_j^{\max} + c_j^{\min}}{2} \right| \leq \frac{c_j^{\max} - c_j^{\min}}{2} + s_j^{\min}(t+k\delta), \\
 & \quad \quad s_j^{\max}(t+\delta) \geq 0, \dots, s_j^{\max}(t+n_h\delta) \geq 0, \\
 & \quad \quad s_j^{\min}(t+\delta) \geq 0, \dots, s_j^{\min}(t+n_h\delta) \geq 0, \\
 & \quad \quad \text{for } j = 1, \dots, N, \text{ and } k = 1, \dots, n_h, \\
 & \quad \quad \mathbf{R}(t), \dots, \mathbf{R}(t+(n_h-1)\delta) \in \mathcal{R},
 \end{aligned} \tag{16}$$

273 where the constraints have the same meaning of Eq. 11.

274 **5. ToBike system description, data analysis, and parameter identification**

275 ToBike has been Turin's (Italy) bike-sharing system since 2010. It comprises
276 more than 160 stations and 2000 bikes. Stations contain a variable number of
277 docking slots. About 8000 trips per day are executed by more than 20,000 cus-
278 tomers within an area of approximately 300 square kilometers. No reservation
279 system is available, but users can use a mobile app to check the number of bikes
280 and docking slots available at a given station. A user that wishes to travel can pick
281 up an available bike at any station, cycle toward a destination station and dock it
282 to an available docking slot. Bike stations are very dense in the urban territory, es-
283 pecially around the city center. Thus, in most of the cases, the availability of bikes
284 is much efficiently and representatively computed over groups of stations, rather
285 than on single ones. Toward the design of an effective control algorithm, stations
286 have therefore been grouped in clusters, according to their geographic location.
287 Clustering has been performed via a k -means algorithm [30] and a Voronoi tes-
288 sellation, partitioning the city territory in $N = 10$ groups of stations, based on the
289 Euclidean distance between them. Such a distance is evaluated on the azimuthal
290 equidistant projection [31] of the geographical coordinates, with the azimuth set
291 on the geometric center of the stations. A similar partition strategy can be used
292 to apply our modeling and control approach to free-floating systems, i.e., systems
293 where vehicles are not obliged to be picked up and dropped off at stations and bal-
294 ancing should be pursued over urban zones to be determined according to given
295 criteria [32].

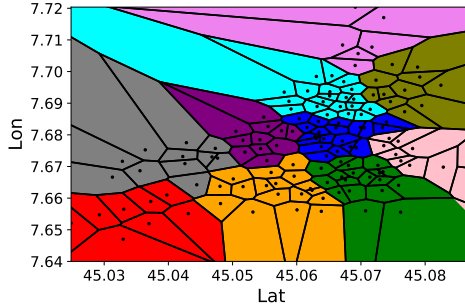


Figure 2: Voronoi tessellation of the urban territory for vehicle rebalancing strategies, obtained through our clustering algorithm. Each point represents a station. The color code indicates the station groups obtained with a k -means algorithm.

296 After the grouping procedure, each selected location contains a different num-
 297 ber of stations. For simulation purposes, the initial number of bikes in each loca-
 298 tion and the location capacity constraints c_j^{\min} , c_j^{\max} are set up to be proportional
 299 to the number of stations that each group contains, denoted as N_j , $j = 1 \dots 10$.
 300 Table 2 summarizes such values for the selected locations.

301 Data analysis confirms the intuition that bike trips have a strong seasonal-
 302 ity along the day, week, and year time-scales, and that there exist recurrent time
 303 periods with a travel activity close to zero. A significant and dynamically rich
 304 dataset is therefore obtained by filtering out from the database the time periods
 305 in which usage is extremely low and can be approximated to zero. In particular,
 306 all the trips occurred in 2015 during all workdays from 8AM to 8PM are con-
 307 sidered, excluding January, February (when bikes are almost not used due to the
 308 low temperatures), and August (holiday month in Italy for the majority of peo-
 309 ple). Furthermore, all the rebalancing operations made by ToBike operators are
 310 removed from the database.

311 A piece-wise constant identification of the parameters is performed, with a

location	N_j	$z_j(0)$	c_j^{\min}	c_j^{\max}
1	8	80	40	120
2	19	190	95	285
3	18	180	90	270
4	6	60	30	90
5	17	170	85	255
6	11	110	55	165
7	20	200	100	300
8	11	110	55	165
9	10	100	50	150
10	11	110	55	165

Table 2: Summary of location parameters: number of stations after the clustering procedure (N_j), initial number of bikes at the beginning of each simulation ($z_j(0)$), lower and upper capacity bounds (c_j^{\min} , c_j^{\max}).

312 time period of $\Delta = 1$ hour. Specifically, $\lambda_i(t)$ is estimated as the average number
313 of bikes per minute that leave location i in the hourly time-window that contains
314 t ; $p_{ij}(t)$ is estimated as the fraction of bikes that leave i and arrive in j in the
315 hourly time-window that contains t . Data analysis leads to the observation that
316 travel durations are not affected by the time of the day. Thus, the dependency on t
317 of $\tilde{q}_{ij}(t, \delta)$ can be neglected, and according to section 2.1.3, such quantity can be
318 estimated as $\tilde{q}_{ij}(\delta) = 1 - e^{-\delta/\tau_{ij}}$, where τ_{ij} is the average transit time over the link
319 (i, j) . This simplification would likely not apply to car sharing systems, where car
320 trip durations are heavily affected by traffic conditions. More details and alternate
321 approaches on the identification of the system parameters can be found in our

322 previous works [23, 24].

323 Due to the mentioned choices in the data analysis phase, a day-long simulation
324 consists of twelve instances of simulation of Eqs. (3)-(4), where parameters are
325 considered to vary in an hourly piece-wise constant fashion and the initial value
326 of state variables in each simulation phase after the first one is set equal to the
327 final value of the preceding one. Figure 3(a) shows the outcome of a Monte Carlo
328 simulation campaign executed over 100 daily independent simulations, illustrating
329 the distribution of the percentage variation of the number of bikes with respect to
330 the initial value $z_i(0)$, in absence of control action, i.e., $r_{ij}(t) \equiv 0, \forall i, j, t$. It can be
331 observed that the system dynamics tends to evolve toward imbalance conditions,
332 with some locations that tend to fill in at the expense of others, which tend to
333 become empty. Figure 3(c) illustrates more in detail this condition, by plotting the
334 trend of the number of bikes in location 4 over a day, that is, $z_4(t)$, highlighting
335 the trend for this location to fill up as long as time progresses.

336 In order to assess the performance of the proposed control algorithms, a stress
337 test is first carried out in open loop, by proportionally increasing the original $\lambda_i(t)$
338 estimated from trip data to the value $\alpha\lambda_i(t)$, where the stress factor $\alpha > 1$, for
339 all stations at all times, simulates a perturbation upon the usual operational con-
340 ditions. Since this is an open loop test, rebalancing inputs are considered null at
341 any time, that is, $r_{ij}(t) \equiv 0, \forall i, j, t$.

342 In the proposed simulation, the stochastic equations (3)-(4) are simulated se-
343 lecting $\alpha = 4$ and generating departures according to Eq. (1). The application
344 of such increased departure rates yields an important imbalance on all locations,
345 as can be observed by comparing Figs. 3(a) and 3(b). It can be observed that
346 even the application of smaller values of the stress factor α to departure rates im-

347 plies important imbalance throughout the stations. To this aim, Fig. 3(d) plots the
 348 trend of the *global imbalance parameter* f_E as a function of the stress factor α .
 349 Such a global imbalance parameter is defined as the average number of bikes, per
 350 time-step δ , that exceeds the prescribed capacity bounds at each location. More
 351 specifically, it is computed as $f_E = \sum_{k=0}^{n_h-1} \sum_j s_j(t+k\delta)/n_h$. It can be observed
 352 that important imbalances are revealed for values of α slightly greater than the
 353 unity, to then grow rapidly as the stress factor α increases.

354 6. Results

355 In this section, we present extensive numerical results to assess the perfor-
 356 mance of the proposed rebalancing strategies. The main performance parameters
 357 here analyzed are summarized in Table 3.

Variable	Description
γ	Trade-off between rebalancing accuracy and control effort
η	Confidence interval for the attainment of the robust control objective
f_E	Random variable expressing the global imbalance per time step
f_R	Random variable expressing the global control effort per time step
$\mathcal{R}_{\text{link}}$	Random variable expressing the total control effort per link
$\mathcal{D}_{\text{link}}$	Random variable expressing the total traffic intensity per link

Table 3: Nomenclature of the main performance parameters used in the paper.

358 6.1. Control of the expected values

359 The first set of numerical experiments concerns the application of the MPC-
 360 based control of the expected values of the state variables, as described in Sec-

361 tion 4.3. Notably, the optimization program (11) is solved to generate the control
 362 inputs, which are in turn applied to Eqs. (3)-(4) to simulate the dynamical sys-
 363 tem response. Fixing a time horizon of n_h time units, the whole control sequence
 364 $\{\mathbf{R}_{ij}(t), \mathbf{R}_{ij}(t + \delta), \dots, \mathbf{R}_{ij}(t + n_h\delta)\}$ is generated solving (11)¹. Then, only the
 365 control action $R_{ij}(t)$ is applied to the simulator in Eqs. (3)-(4). At each time step,
 366 the simulator selects the number of bikes $d_{ij}(t)$ moving from i to j according to a
 367 Poisson process with expected value $\lambda_i(t)p_{ij}(t)$ plus the control operation $R_{ij}(t)$.
 368 The number of bikes arriving from i to j are selected via a probabilistic round-
 369 ing [33] of $q_{ij}(\delta)v_{ij}(t - 1)$. Here and henceforth, the whole simulation time is
 370 fixed to $T = 12$ hours, and the time-step to $\delta = 15$ minutes, unless differently
 371 specified. The MPC control strategy is executed over a shrinking observation
 372 window with duration $T = 12$ hours, and with a prediction horizon $n_h = 48$ time
 373 units, that is, 12 hours.

374 An extensive Monte Carlo simulation campaign is executed to validate the
 375 proposed approach, achieving satisfactory results. One hundred independent sim-
 376 ulations are performed, randomized in the realizations of the vehicle departures,
 377 starting from the initial conditions of Table 2. Figure 4 exemplifies the obtained
 378 results, illustrating the trend of the occupancy of two locations (variables z_4 and
 379 z_7) over a 12 hours interval, from 8AM to 8PM, in uncontrolled and controlled
 380 situations, respectively. The two locations have different spatial characteristics:
 381 location 4 is peripheral, whereas location 7 is central. As can be observed from
 382 the figure, in uncontrolled mode location 4 tends to become empty at the begin-
 383 ning of the workday, whereas it tends to fill up later on. Location 7, as expected,

¹We remark that the actual length of the prediction horizon n_h can eventually change in time,
 due to the shrinking of the remaining simulation time.

384 exhibits the opposite behavior. Both occupancy values are clearly outside the
385 fixed desired occupancy, which would allow an efficient usage of the system, see
386 Figs. 4 (a) and (c). The application of the proposed control strategy with a trade-
387 off parameter set as $\gamma = 0.1$ leads to the results illustrated in Figs. 4 (b) and (d). It
388 can be observed that, for both locations, the desired control performance in terms
389 of the average value of state variables is attained. However, the effects of stochas-
390 ticity cannot always be neglected. The average value of z_7 , in fact, lays in a central
391 region within the two prescribed bounds and, as a consequence, stochastic varia-
392 tions are also contained within those bounds. On the other hand, the control action
393 tends to keep the average value of z_4 very close to the lower bound, implying that
394 stochastic fluctuations lead to bound violations for about half of the trials.

395 This issue is successfully tackled in the next Section, where the results of
396 the robust control strategy based on interval modeling defined in Section 4.4 are
397 reported.

398 *6.2. MPC control based on robust interval modeling*

399 In this Section, numerical results obtained using the MPC control based on
400 robust interval modeling, defined in Section 4.4, are reported. Similarly to the
401 previous section, the whole control sequence over a time horizon of $n_h = 48$ time
402 units is obtained by solving the optimization program (16), then only the current
403 control input is applied to the system, the rest of the sequence is discarded, the
404 system is simulated using Eqs. (3)-(4), the time index is updated and the control
405 loop resumed.

406 Two parameters are important to assess the performance of the proposed con-
407 trol strategy: the trade-off parameter γ and the probability η , which in turn defines
408 probabilistic bounds for the state variables (see Section 6.2). Besides the global

409 imbalance parameter f_E previously defined, a parameter representative of the con-
410 trol effort is considered, that is, the average number of bikes displaced for control
411 purposes per time-step δ , computed as $f_R = J_T/n_h$.

412 The first assessment of the system performance consists of the evaluation of
413 the two performance parameters as a function of the two control parameters. To
414 this aim, the contour plots of Fig. 5, obtained performing 10 independent simula-
415 tions for pairs of values of η and γ , selected over a two-dimensional grid spanning
416 20 values for each variable. The control effort is illustrated in Fig. 5(a). As in-
417 tuition may suggest, directly from inspection of problem (16), larger values of γ
418 imply a lower control effort. Parameter η has not, in general, a great influence
419 on the control effort. However, its main role is to set a feasibility threshold on
420 the control problem solution, and this threshold monotonically changes with η .
421 Fixing a given value for η , in fact, implies fixing an estimate of the stochastic
422 fluctuation of the state variables with a certain probability. Fixing a value for η ,
423 increasing γ implies the willingness to solve the control problem with a lower
424 control effort. When the prescribed control effort is too small, problem (16) be-
425 comes unfeasible and the control objective cannot be attained. Increasing η , on
426 the other hand, implies to consider greater stochastic fluctuations. Therefore, the
427 limit for which the control problem is unfeasible is reached for lower values of γ .
428 Figure 5(b), on the other hand, is representative of the rebalancing performance.
429 It can be observed that, in general and as expected, a high control effort implies
430 an excellent balancing performance. Focusing on the rebalancing parameter f_E
431 alone, it can be observed that for intermediate values of the trade-off parameter
432 (between about 0 and 6), f_E has a minimum for fixed γ , depending on η . This
433 implies that there exists a range for the confidence interval of stochastic variabil-

434 ity that yields the best performance. This can be explained considering that, for
435 very low values of η the control reduces to the control of the median value of the
436 state variable, whereas, on the other hand, high values of η tend to overestimate
437 the stochastic variability with the only effect of increasing the control effort and
438 decreasing the control performance. For high values of γ , on the other hand, the
439 performance with respect to η is monotonically decreasing. This is due to the fact
440 that the main aim is to control the system assuming a high stochastic fluctuation
441 and a low control effort, yielding unsatisfactory performance.

442 Figure 6 offers more elements of assessment and comparison with the previous
443 control technique. For comparison purposes, Fig. 6(a) plots the two performance
444 parameters, f_R vs. f_E . Each curve is plotted as a function of the trade-off param-
445 eter γ , which varies in logarithmic steps in the interval $\gamma \in [0.1, 100]$, for three
446 fixed values of the confidence probability η : 0.01, 0.5, and 0.99. The following
447 observations are in order: *i*) increasing the values of γ generally improves the sys-
448 tem performance, both in terms of control effort and in terms of system balancing,
449 however, this effect tend to saturate and further increases of γ become practically
450 ineffective; and *ii*) the method based on robust interval modeling improves the
451 performance of the control of the expected value for intermediate values of η ,
452 implying that there exists an *optimal value* for η , which guarantees the best im-
453 provement with respect to the control of the expected value. Figure 6(b) offers
454 a direct insight on the occupancy state z_4 of location 4. Results are obtained in
455 the same conditions of the control of the expected value (see Fig. 4(b)) and for
456 $\eta = 0.5$. It can be observed that the control technique based on robust interval
457 modeling outperforms the control technique based on expected values practically
458 everywhere. However, a limited number of realizations of the simulations still

459 yield an occupancy status slightly out of the prescribed bounds. This is an effect
460 of stochasticity, which can be hardly totally eradicated.

461 More insight on the rebalancing effort is offered in Fig. 7. Specifically, in
462 Fig. 7(a), the trend of the number of bikes that travel in the system is plotted
463 versus time. Namely, the number of bikes that are displaced by users' trips is
464 displayed along with the corresponding number of bikes displaced for rebalanc-
465 ing purposes, for both the control on the expected value and the control based on
466 robust interval modeling. Most importantly, it is observed that the rebalancing
467 traffic is much less than the users' traffic, implying that the control is efficient.
468 An observation of the opposite phenomena, on the other hand, would imply that
469 almost every trip executed by a user should be compensated by a rebalancing trip,
470 which is extremely costly and not in the spirit of a MOD system. A detailed il-
471 lustration of the rebalancing trips in time is given in Fig. 7(b). There, it can be
472 observed that the rebalancing activity using robust interval modeling is higher and
473 has a greater variance than the control of the expected value, in order to compen-
474 sate for the state variable stochasticity. Also, the rebalancing activity presents an
475 important peak, corresponding to an intense rebalancing activity to compensate
476 for the commuters' trips from peripheral to central locations at the start of the
477 working day. Then, a lower and almost steady activity is realized, to compensate
478 for the reduced number of trips. A slight increase toward the end of the work-
479 ing day tends to compensate for the traffic imbalance from the city center to the
480 peripheral areas. The peak at the end of the day is less evident than that at the
481 beginning of the day, since the system is more balanced by the predictive rebal-
482 ancing activity throughout the day. It can be verified that the intra-day pattern of
483 user trips is scarcely correlated with the number of rebalancing operations. This

484 counterintuitive observation implies that the rebalancing operations are not very
 485 dependent on the traffic, and this is due to the prediction capabilities of the model.
 486 The morning peak in the rebalancing operations, in fact, is present because the
 487 algorithm cannot be very accurate when it is initialized and starts the prediction.
 488 Figures 7(c) and 7(d) focus on the activity over the network links, that is, over the
 489 routes that connect one station to another. Figure 7(c) illustrates the cumulative
 490 distribution function (CDF) of the random variable $\mathcal{R}_{\text{link}}$, obtained summing all
 491 the rebalancing operations per link, over the entire time span and averaged over
 492 100 independent simulations, for both the control of the expected value and that
 493 based on robust interval modeling. Interestingly, while the control based on ro-
 494 bust interval modeling overall performs more rebalancing actions than that based
 495 on the expected value, these actions are concentrated on a fewer number of links.
 496 In fact, it can be verified that around the 40% of links experience a control ac-
 497 tivity close to zero in the robust control case. On the other hand, the control of
 498 the expected value performs rebalancing actions on almost all links of the system
 499 during one day. In conclusion, the control based on robust interval modeling is
 500 more efficient than that based on the expected value. Figure 7(d) plots the CDF of
 501 the random variable $\mathcal{D}_{\text{link}}$, obtained summing all the user trips per link, over the
 502 entire time span and averaged over 100 independent simulations. The CDF plot
 503 of Fig. 7(d) confirms that user traffic is much higher than the rebalancing activity
 504 and that it equally occurs over most of the link during the day.

505 Figure 8 focuses on the performance assessment with respect to local perturba-
 506 tions of the user demand, in space and time. The figures are obtained by selecting
 507 a random number ν of pairs of locations and time instants (i, t) , and by impos-
 508 ing an instantaneous increment of the user demand $\lambda_i(t)$. Selected instantaneous

509 increments for this set of simulations are 10% and 20%. Figure 8(a) illustrates
510 the result obtained with the control of the expected value, while Fig. 8(b) refers
511 to the control based on robust interval modeling. As intuition may suggest, with
512 both control strategies the imbalance parameter f_E tends to increase when the de-
513 mand increase is more frequent in time and space. It can also be observed that
514 the trend in the detriment of performance is almost linear with the number of per-
515 turbations. However, the performance attained with the control based on robust
516 interval modeling is better than that attained through the control of the expected
517 value.

518 The last set of simulations in Fig. 9 deal with a reduction of the traffic speed
519 over a randomly selected fraction of links. In particular, the value of variable
520 $q_{ij}(\delta)$ over the selected links is reduced by 20%, that is, $q_{ij}(\delta) \rightarrow 0.8 q_{ij}(\delta)$. It can
521 be observed that, while the control of the expected value improves its performance
522 with a slow down of a growing number of links (Fig. 9(a)), the performance of the
523 control based on robust interval modeling is nearly not affected by such a variation
524 (Fig. 9(b)). This is due to the fact that the control of the expected value is more
525 prone to compensate for stochasticity, which is not explicitly accounted for, if it
526 has more time available to implement the control strategy. On the other hand,
527 the control based on robust interval modeling explicitly accounts for stochasticity,
528 and the time available to compensate occurring variations is not relevant.

529 *6.3. Investigation on seasonality patterns and the role of the time-horizon*

530 In this section, we investigate on the periodicity of the system dynamics and
531 the sensitivity of the performance of the control strategy with respect to the length
532 of the MPC time horizon n_h . Here, we use a simulation time of a workweek, i.e.,
533 5 consecutive days. Our simulations start at midnight on the first day and end at

534 midnight on the fifth day. To be consistent with the previous sections, and in view of
535 the fact that no substantial differences in the system parameters have been revealed
536 over different workdays, the input traffic parameters, $\lambda_i(t)$ and $p_{ij}(t)$ are repeated
537 each day until the end of the simulation, whereas the rest of the quantities are
538 simulated over a week through the stochastic model. The first simulation regards
539 the effect of the MPC time horizon on the violation of the capacity constraints. In
540 Fig. 10(a), we compare the outcome of two sets of Monte Carlo simulations, ex-
541 ecuted by applying the robust interval control method with an MPC time horizon
542 of 2 and 12 hours, respectively. We plot the difference between the average values
543 of the slack variables computed using the two time horizons. Such variables, we
544 recall, quantify the number of violations of capacity constraints. We observe a
545 slight trend whereby the 12-hours time horizon is advantageous over the 2-hours
546 one, and such an advantage increases as long as the simulation time increases.
547 However, the advantage is contained in the order of magnitude of fractions of ve-
548 hicles per time unit. A more evident advantage of longer time-horizons resides
549 in the magnitude of the control effort. Figure 10(b) illustrates the trend of the
550 global control effort f_R , computed over Monte Carlo simulations, as a function
551 of the MPC time horizon. In this case, the advantage in terms of control effort is
552 apparent, since it monotonically decreases with increasing length of the the time
553 horizon. We report that no relevant differences are observed applying the control
554 of the nominal values, hence, the related results are not reported here for brevity.
555 The most advantageous time horizon of 12 hours has been used to study the sys-
556 tem behavior over a working week. Notably, Fig. 10(c) illustrates the distribution
557 densities of the overall customer departures and overall rebalancing activity. We
558 observe that the proposed model concentrates most of the rebalances after the

559 morning utilization peak, and overnight. Other times in the day involve a very
560 moderate rebalancing action. This implies that efficient rebalancing can be per-
561 formed with two different means: during the peak, with trucks, able to displace
562 a high number of vehicles in a short time. Off-peak, on the other hand, the low-
563 intensity rebalancing can be executed using smaller vehicles, like carts, or through
564 incentives to users.

565 **7. Conclusions**

566 In this work, a novel control-oriented model able to describe the dynamics of
567 MOD systems, either station-based or using floating locations, is introduced. The
568 model enables analytical tractability, since it is described with dynamical equa-
569 tions in the state-space, using stochastic state variables. Reasonable simplifying
570 assumptions lead to the definition of two MPC-based control strategies. First, a
571 control of the expected value of the state variables is devised. Then, a robust con-
572 trol strategy that explicitly accounts for stochastic fluctuations via robust interval
573 modeling is proposed. The model has been validated and its performance assessed
574 on a real dataset of logged trips, made available by ToBike, one of the bike-sharing
575 providers in Turin, Italy. Model parameters can be identified with relatively sim-
576 ple data processing operations. Both control methods lead to satisfactory results
577 with a relatively low control effort. The slightly higher computational complex-
578 ity of the robust controller based on interval modeling is fully repaid by a more
579 efficient performance under different point of views. First, the control attains the
580 expected results in most of the cases, despite the presence of important stochas-
581 tic fluctuations. Second, even though the robust control usually displaces more
582 vehicles than the control based on the expected values, those vehicles are moved

583 among a smaller set of routes, implying a control action that is more spatially
584 concentrated. This obviously helps the design of rebalancing policies based on
585 the displacement of large quantities of vehicles using trucks. Finally, the robust-
586 ness of the proposed control methods to randomly applied perturbations and traffic
587 slowdowns has been assessed, observing a gentle degradation of the performance,
588 which makes the proposed methods promising for applications to real cases.

589 Future directions in our research contemplate the realization of optimized re-
590 balancing strategies based on the rebalancing flows given by the solution of the
591 optimization problems defined in this work. The typical rebalancing pattern ob-
592 tained in this study exhibits a strong peak in the early morning and a less intense,
593 and almost steady rebalancing activity throughout the day. This observation sug-
594 gests that mixed rebalancing policies based on massive displacements in the first
595 hours of the morning, e.g. operated by high-capacity trucks, followed by human-
596 based rebalancing with less volume and frequency, e.g., achieved through incen-
597 tives or gamification, may constitute viable and efficient strategies to make MOD
598 systems effective, efficient, and available at any time of the day, increasing both
599 the customer satisfaction and the provider revenues.

600 **Acknowledgements**

601 This research is supported by Compagnia di San Paolo, Italy. The authors
602 warmly acknowledge Bicincittà-ToBike for providing logged service data.

603 **Conflict of Interest**

604 The authors declare that they have no conflict of interest.

- 605 [1] R. Meddin, P. DeMaio, The bike-sharing world map, <http://www.bikesharingworld.com/>, 2018.
- 606
- 607 [2] Car2Go, <https://www.car2go.com/NA/en/press/>, accessed: 2018-02-28, ????
- 608
- 609 [3] P. Borgnat, P. Abry, P. Flandrin, C. Robardet, J.-B. Rouquier, E. Fleury, Shared bicycles in a city: A signal processing and data analysis perspective, *Advances in Complex Systems* 14 (03) (2011) 415–438.
- 610
- 611
- 612 [4] J. Froehlich, J. Neumann, N. Oliver, Measuring the pulse of the city through shared bicycle programs, *Proceedings of UrbanSense08* (2008) 16–20.
- 613
- 614 [5] R. Nair, E. Miller-Hooks, R. C. Hampshire, A. Bušić, Large-scale vehicle sharing systems: analysis of Vélib’, *International Journal of Sustainable Transportation* 7 (1) (2013) 85–106.
- 615
- 616
- 617 [6] M. Nourinejad, S. Zhu, S. Bahrami, M. J. Roorda, Vehicle relocation and staff rebalancing in one-way carsharing systems, *Transportation Research Part E: Logistics and Transportation Review* 81 (2015) 98–113.
- 618
- 619
- 620 [7] Z. Yang, J. Hu, Y. Shu, P. Cheng, J. Chen, T. Moscibroda, Mobility Modeling and Prediction in Bike-Sharing Systems, in: *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, ACM, 165–178, 2016.
- 621
- 622
- 623
- 624 [8] E. O’Mahony, D. B. Shmoys, Data Analysis and Optimization for (Citi) Bike Sharing., in: *AAAI*, 687–694, 2015.
- 625

- 626 [9] M. Benchimol, P. Benchimol, B. Chappert, A. De La Taille, F. Laroche,
627 F. Meunier, L. Robinet, Balancing the stations of a self service bike hire
628 system, *RAIRO-Operations Research* 45 (1) (2011) 37–61.
- 629 [10] D. Chemla, F. Meunier, R. W. Calvo, Bike sharing systems: Solving the
630 static rebalancing problem, *Discrete Optimization* 10 (2) (2013) 120–146.
- 631 [11] M. Rainer-Harbach, P. Papazek, B. Hu, G. R. Raidl, Balancing bicycle
632 sharing systems: A variable neighborhood search approach, in: *European
633 Conference on Evolutionary Computation in Combinatorial Optimization*,
634 Springer, 121–132, 2013.
- 635 [12] M. Dell’Amico, E. Hadjicostantinou, M. Iori, S. Novellani, The bike sharing
636 rebalancing problem: Mathematical formulations and benchmark instances,
637 *Omega* 45 (2014) 7–19.
- 638 [13] M. Dell, M. Iori, S. Novellani, T. Stützle, et al., A destroy and repair al-
639 gorithm for the bike sharing rebalancing problem, *Computers & Operations
640 Research* 71 (2016) 149–162.
- 641 [14] G. Erdoğan, M. Battarra, R. W. Calvo, An exact algorithm for the static
642 rebalancing problem arising in bicycle sharing systems, *European Journal of
643 Operational Research* 245 (3) (2015) 667–679.
- 644 [15] A. Faghih-Imani, R. Hampshire, L. Marla, N. Eluru, An empirical analysis
645 of bike sharing usage and rebalancing: Evidence from Barcelona and Seville,
646 *Transportation Research Part A: Policy and Practice* 97 (2017) 177–191.

- 647 [16] T. Raviv, M. Tzur, I. A. Forma, Static repositioning in a bike-sharing sys-
648 tem: models and solution approaches, *EURO Journal on Transportation and*
649 *Logistics* 2 (3) (2013) 187–229.
- 650 [17] C. Contardo, C. Morency, L.-M. Rousseau, Balancing a dynamic public
651 bike-sharing system, vol. 4, Cirrelt Montreal, 2012.
- 652 [18] J. Schuijbroek, R. C. Hampshire, W.-J. Van Hoes, Inventory rebalancing
653 and vehicle routing in bike sharing systems, *European Journal of Operational*
654 *Research* 257 (3) (2017) 992–1004.
- 655 [19] D. Chemla, F. Meunier, T. Pradeau, R. W. Calvo, H. Yahiaoui, Self-
656 service bike sharing systems: simulation, repositioning, Tech. Rep., pric-
657 ing. Tech. Rep. hal-00824078, Centre dEnseignement et de Recherche en
658 Mathématiques et Calcul Scientifique (CERMICS), 2013.
- 659 [20] J. Pfrommer, J. Warrington, G. Schildbach, M. Morari, Dynamic vehicle re-
660 distribution and online price incentives in shared mobility systems, *IEEE*
661 *Transactions on Intelligent Transportation Systems* 15 (4) (2014) 1567–
662 1578.
- 663 [21] A. Singla, M. Santoni, G. Bartók, P. Mukerji, M. Meenen, A. Krause, Incen-
664 tivizing Users for Balancing Bike Sharing Systems., in: *AAAI*, 723–729,
665 2015.
- 666 [22] C. Fricker, N. Gast, Incentives and redistribution in homogeneous bike-
667 sharing systems with stations of finite capacity, *Euro journal on transporta-*
668 *tion and logistics* 5 (3) (2016) 261–291.

- 669 [23] G. C. Calafiore, F. Portigliotti, A. Rizzo, A Network Model for an Urban
670 Bike Sharing System, *IFAC-PapersOnLine* 50 (1) (2017) 15633–15638.
- 671 [24] G. C. Calafiore, F. Portigliotti, A. Rizzo, C. Novara, A flow optimization
672 approach for the rebalancing of mobility on demand systems, in: *IEEE Con-*
673 *ference on Decision and Control*, 1–6, 2017.
- 674 [25] D. Zeng, J. Xu, J. Gu, L. Liu, G. Xu, Short term traffic flow prediction based
675 on online learning SVR, in: *Power Electronics and Intelligent Transportation*
676 *System, 2008. PEITS'08. Workshop on, IEEE*, 616–620, 2008.
- 677 [26] L. Zhao, F.-Y. Wang, Short-term traffic flow prediction based on ratio-
678 median lengths of intervals two-factors high-order fuzzy time series, in: *Ve-*
679 *hicular Electronics and Safety, 2007. ICVES. IEEE International Conference*
680 *on, IEEE*, 1–7, 2007.
- 681 [27] I. Kaysi, M. E. Ben-Akiva, H. Koutsopoulos, An integrated approach to ve-
682 hicle routing and congestion prediction for real-time driver guidance, vol.
683 1408, *Transportation Research Board*, 1993.
- 684 [28] G. Calafiore, C. Bongiorno, A. Rizzo, A control-oriented model for mobil-
685 ity on demand systems, in: "Proceedings of the 57th IEEE Conference on
686 Decision and Control, CDC 2018", 2018.
- 687 [29] E. F. Camacho, C. B. Alba, *Model predictive control*, Springer Science &
688 Business Media, 2013.
- 689 [30] S. Lloyd, Least squares quantization in PCM, *IEEE transactions on informa-*
690 *tion theory* 28 (2) (1982) 129–137.

- 691 [31] J. P. Snyder, *Flattening the earth: two thousand years of map projections*,
692 University of Chicago Press, 1997.
- 693 [32] S. Weigl, K. Bogenberger, Relocation strategies and algorithms for free-
694 floating car sharing systems, *IEEE Intelligent Transportation Systems Mag-*
695 *azine* 5 (4) (2013) 100–111.
- 696 [33] S. Gupta, A. Agrawal, K. Gopalakrishnan, P. Narayanan, Deep learning with
697 limited numerical precision, in: *International Conference on Machine Learn-*
698 *ing*, 1737–1746, 2015.

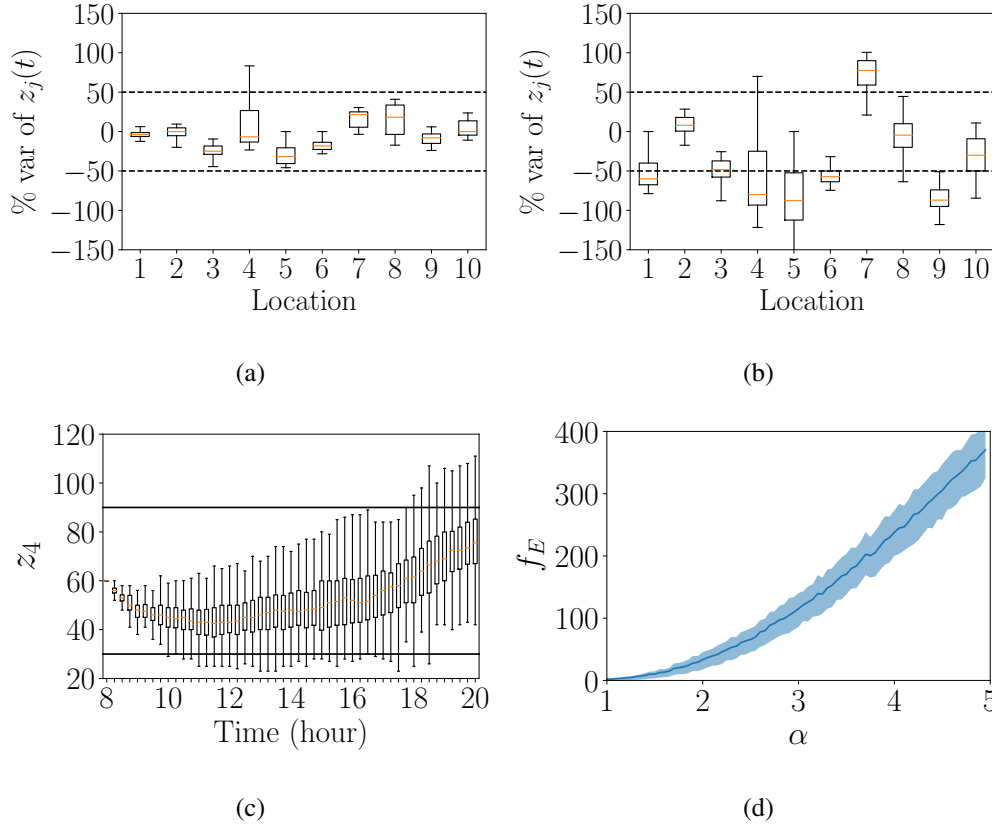


Figure 3: Outcome of an open-loop stress test on the departure rates $\lambda_i(t)$ performed over 100 independent daily Monte Carlo simulations, without the application of control actions. (a) percentage variations, with respect to their initial values $z_i(0)$, of the number of bikes at considered locations, with the original departure rates $\lambda_i(t)$ estimated from logged trip data; (b) same as (a), under the stress test with departures rate proportionally increased by a stress factor $\alpha = 4$; (c) detailed trend, for zone 4, of the state variable $z_4(t)$ with original departure rates. Boxes represent the 25-75 percentile range, whiskers represent extreme values, and orange lines the median ones. Continuous horizontal lines indicate the location capacity limits. In (d), trend of the global imbalance parameter f_E as a function of the stress factor α . The blue line indicates the mean value, the band indicates the 25-75 percentile range.

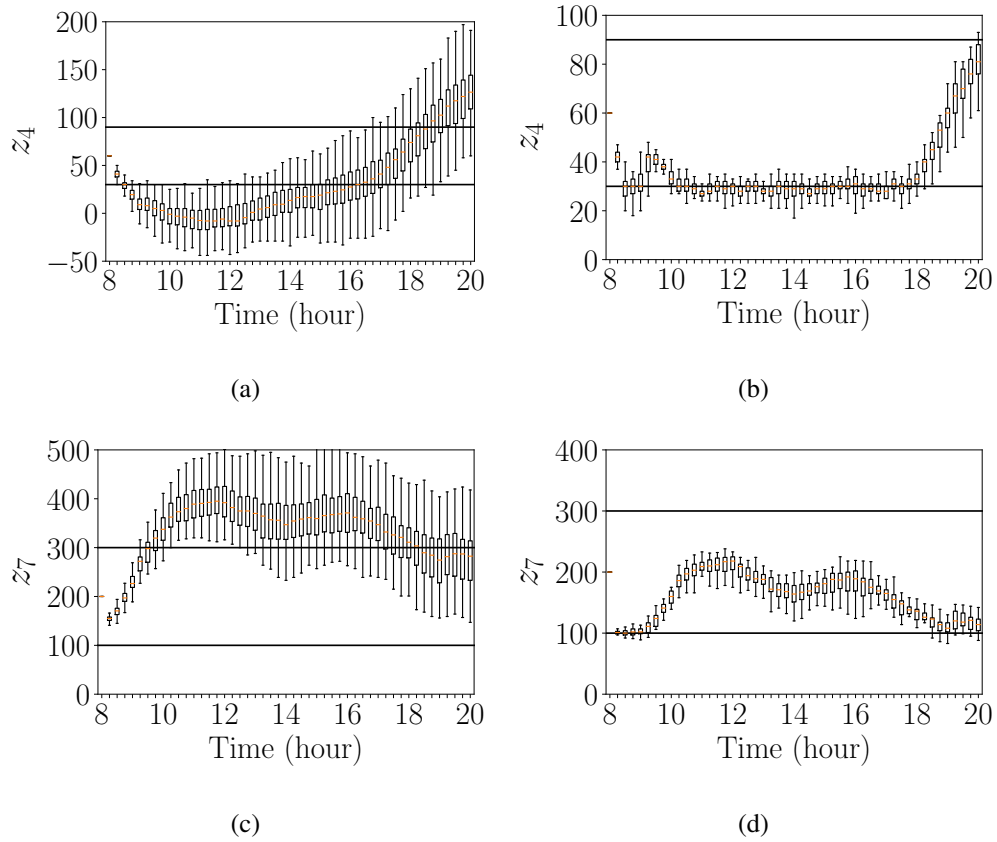


Figure 4: Monte Carlo simulations of the ToBike system in a 12 hours time interval, from 8AM to 8PM. Results are obtained over 100 independent simulations. (a),(b) occupancy for location 4 and location 7, respectively, in uncontrolled mode. (c),(d) simulation for with control on the expected values for the same locations. The continuous upper and lower lines in each plot indicate the desired upper and lower capacity bound for the locations. Boxes represent the 25 – 75 percentile range, whiskers represent extreme values, and orange lines the median ones.

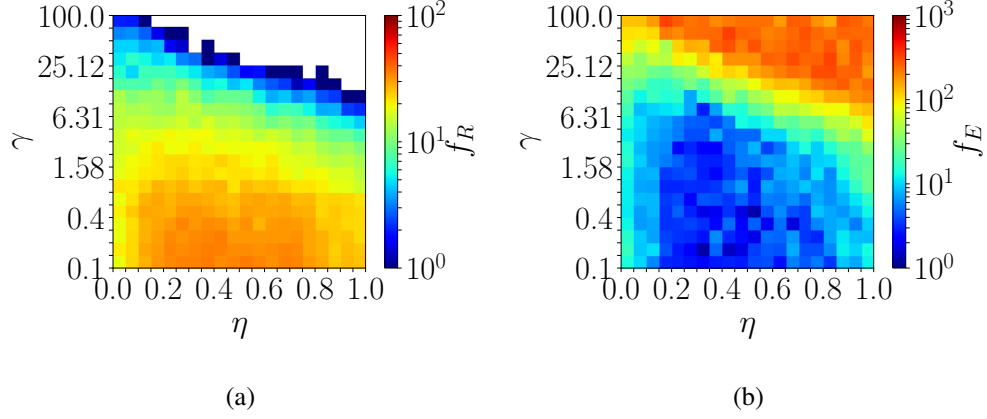


Figure 5: Performance of the MPC control based on robust interval modeling: (a) global control effort as a function of η and γ ; (b) global imbalance indicator for the same parameters. Each value is a mean of 10 independent simulations with a robust control. White regions in panel (a) corresponds to unfeasible solutions of the optimization problem.

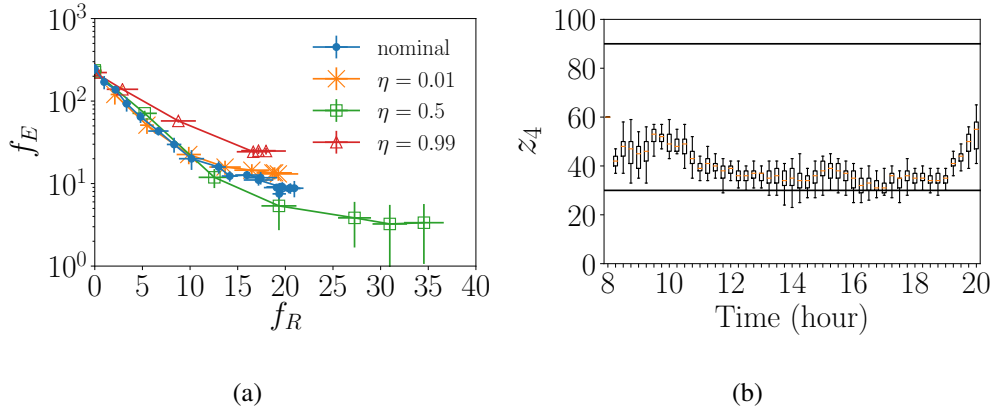


Figure 6: Performance assessment and comparison of the control technique based on robust interval modeling. In (a), performance curves f_E vs. f_R , plotted for a logarithmic interval of $\gamma \in [0.1, 100]$, and for $\eta = 0.01, 0.5, 0.99$. The blue plot corresponds to the performance of the control method based on the expected value, described in Section 6.1. In (b), trend of the occupancy state z_4 over a 12 hours interval, from 8AM to 8PM. Results are obtained in the same condition of the control of the expected value and setting $\eta = 0.5$.

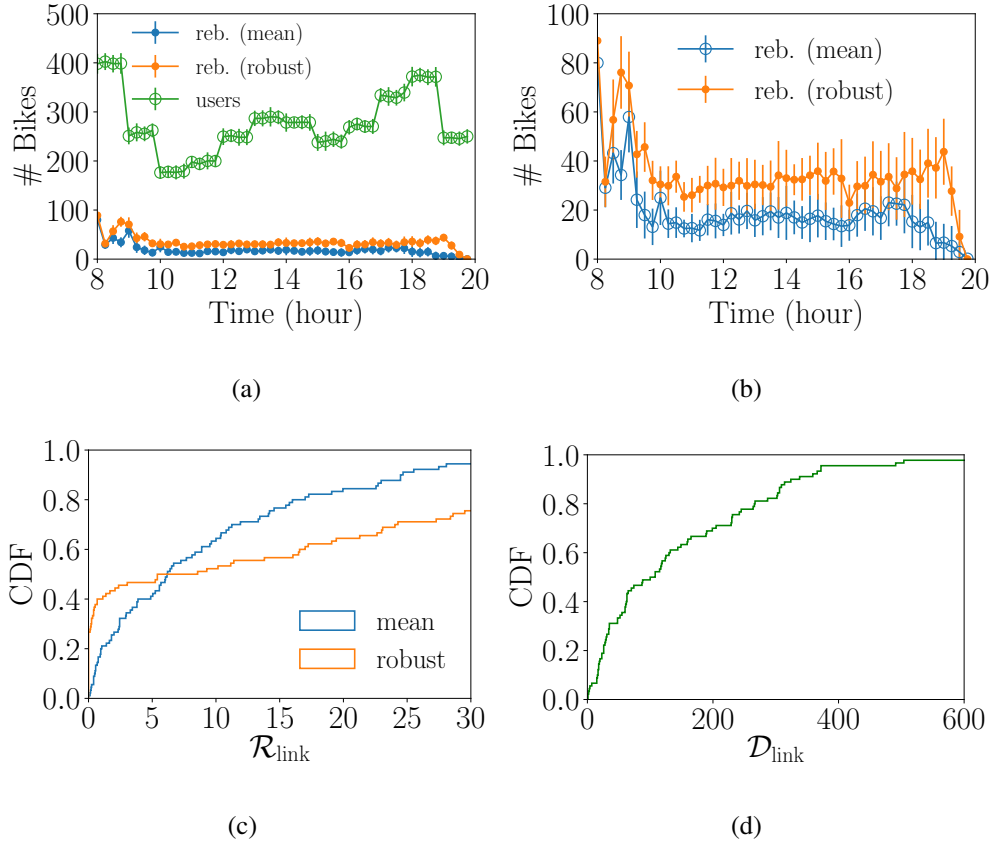


Figure 7: Control performance assessment performed over 100 independent simulations, for $\gamma = 0.1$ and $\eta = 0.5$. In panel (a), the green line illustrates the average number of user trips over time during a whole day, from 8AM to 8PM. The orange and the blue lines illustrate the average number of rebalancing operations during the same day, using the control of the expected value and that based on robust interval training, respectively. Panel (b) illustrates the control activity only, where the vertical bars indicate one standard deviation. Panel (c) illustrates the CDF of the random variable $\mathcal{R}_{\text{link}}$, representative of the control activities over the network links. Panel (d) illustrates the CDF of the random variable $\mathcal{D}_{\text{link}}$, representative of the user trips over the network links.

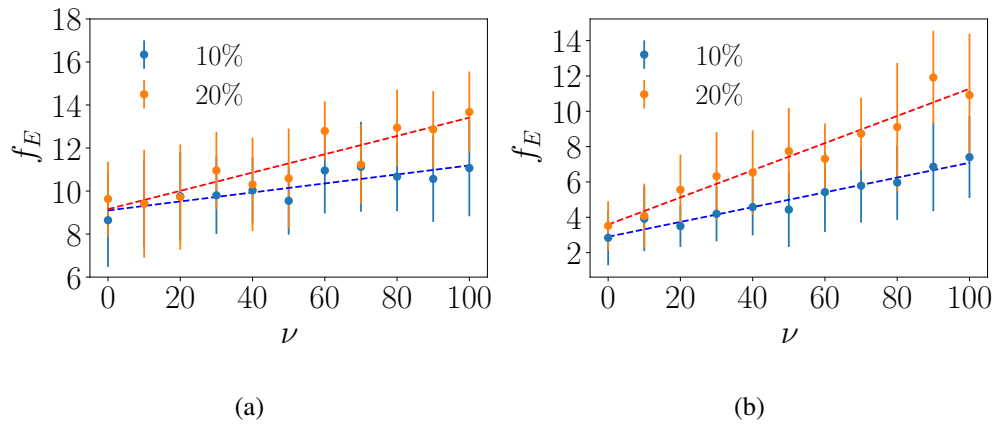


Figure 8: Control performance under instantaneous variations of user demand in time and space. The figures show the effect on global imbalance parameter (f_E) caused by an instantaneous percentage increment of the departure rate that affect a randomly selected subset of size ν consisting of pairs of stations and time instants. Blue dots refer to a demand increment of 10%, orange dots refer to an increment of 20%. Results are obtained averaging over 30 independent simulations. Dots indicate the mean, bars indicate one standard deviation, the dotted line is a linear fit. Panel (a) refers to the control of the expected value, panel (b) to the control based on robust interval modeling.

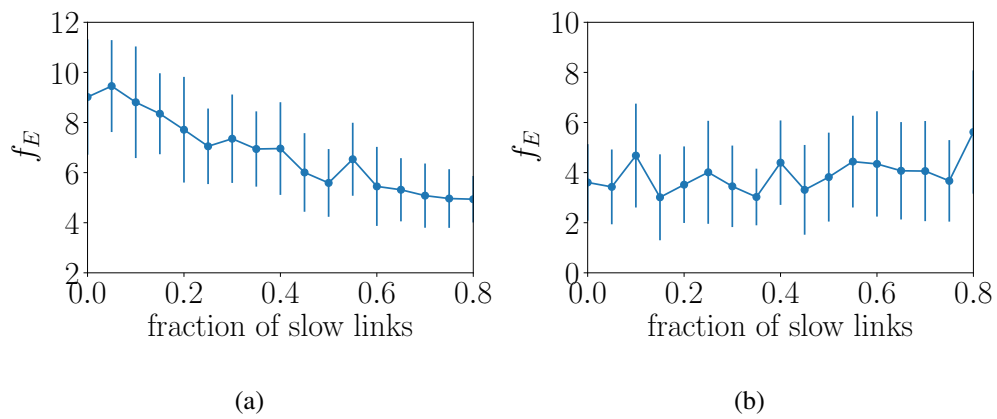


Figure 9: Effects of a local reduction of 20% of the link speed imposed to a random fraction of links. In panel (a), results are obtained via the application of the control of the expected value; in panel (b), via the application of the control with robust interval modeling. Each point is obtained as the average of 30 independent Monte Carlo simulations. Bars indicate one standard deviation.

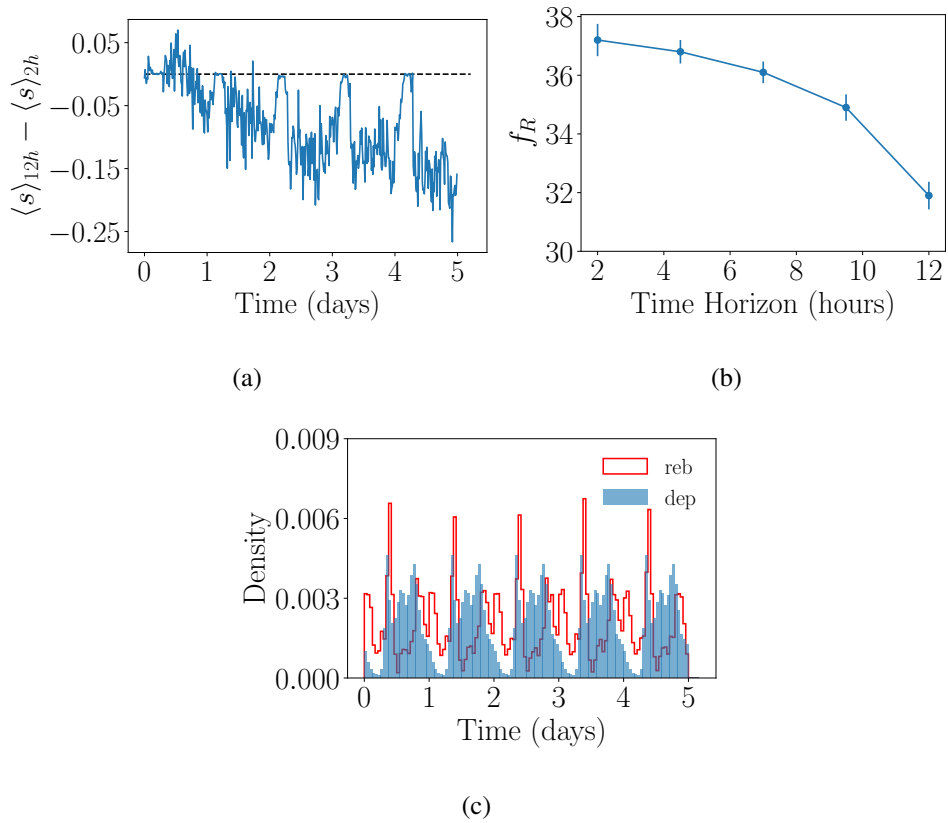


Figure 10: System characterization with respect to the MPC time horizon and weekly periodicity. Results are averaged over 10 independent Monte Carlo simulations. (a) Difference between the overall violation of the capacity constraints with a 2- and 12-hour time horizon; (b) Global control effort f_R as a function of the time-horizon (the bands quantifies the 95% confidence interval); (c) Distribution of the rebalancing operations (red) and expected number of departures (blue) as a function of time, along a workweek, using a 12-hours time horizon.