Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

European Journal of Operational Research xxx (xxxx) xxx

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Innovative Applications of O.R.

On the impact of resource relocation in facing health emergencies

Michele Barbato, Alberto Ceselli, Marco Premoli*

Department of Computer Science, Università degli Studi di Milano, 18, via Celoria, 20133, Milano, Italy

ARTICLE INFO

Article history:
Received 28 July 2021
Accepted 13 November 2022
Available online xxx

Keywords:
OR in health services
COVID-19
Facility location-allocation
Mathematical programming

ABSTRACT

The outbreak of SARS-CoV-2 and the corresponding surge in patients with severe symptoms of COVID-19 put a strain on health systems, requiring specialized material and human resources, often exceeding the locally available ones. Motivated by a real emergency response system employed in Northern Italy, we propose a mathematical programming approach for rebalancing the health resources among a network of hospitals in a large geographical area. It is meant for tactical planning in facing foreseen peaks of patients requiring specialized treatment. Our model has a clean combinatorial structure. At the same time, it considers the handling of patients by a dedicated home healthcare service, and the efficient exploitation of resource sharing. We introduce mathematical programming heuristic based on decomposition methods and column generation to drive very large-scale neighborhood search. We evaluate its embedding in a multi-objective optimization framework. We experiment on real world data of the COVID-19 in Northern Italy during 2020, whose aggregation and post processing is made openly available to the community. Our approach proves to be effective in tackling realistic instances, thus making it a reliable basis for actual decision support tools.

1. Introduction

In the last two decades epidemics of several infectious diseases emerged with increased frequency, in some cases reaching continental or global scale (Jain, Duse, & Bausch, 2018). This is due to a combination of factors, such as proximity of urban areas and wildlife (Cunningham, Daszak, & Wood, 2017), globalization of tourism and of the commercial exchanges (McMichael, 2013), climate changes (McMichael, 2013; Shuman, 2010). The harm posed by epidemics is primarily measured in terms of human losses and health consequences; moreover, responses to epidemics emergencies typically entail high societal and economic costs (see Nicola et al., 2020 for an account of diverse socio-economic consequences of the recent COVID-19 epidemic in high-income countries). These considerations explain the increasing attention of national and international public health agencies toward science-based management of epidemic emergencies. In this context, Operations Research (OR) plays a central role, providing effective methods to support decision-making in epidemics control (Silal et al., 2021) and more generally in large-scale emergencies (Stilianakis & Consoli, 2013).

The outbreak of SARS-CoV-2 and the corresponding surge in patients with severe symptoms of COVID-19 put a strain on health systems. The treatment of COVID-19 patients requires specialized material and human resources, often exceeding the available amount. To provide them with the best possible service, a relocation of health resources between areas in need and areas with a surplus may be put in place.

In this paper we propose a mathematical programming approach for rebalancing the health resources among hospitals of a large geographical area to face a foreseen surge in incoming patients requiring specialized treatment. The rebalancing includes the repurposing of hospital wards, the relocation of medical staff and medical resources, the assignment of incoming patients to suitable wards, the relocation of inpatients between wards, the allocation of medical resources available at external suppliers, and as a last resort the selective discharge of mild patients, possibly through dedicated home healthcare services. Our idea is illustrated in Fig. 1, and we later formalize the overall problem as a mixed-integer linear programming (MILP). Its core stems from the so-called facility location-allocation problems, which arise not only from epidemics management but also from industrial applications: actually, the rebalancing problem described above extends and combines several features of existing facility location-allocation variants and, as such, it has not been treated before.

The problem considered in this paper has relevance at various stages, as it asks to improve the preparedness of the health system in the face of a long-lasting epidemic. We focus on the tactical aspect: the solutions of our methods let decision makers (a) perform a scenario-based validation of the health system resistance

* Corresponding author.
E-mail addresses: michele.barbato@unimi.it (M. Barbato), alberto.ceselli@unimi.it (A. Ceselli), marco.premoli@unimi.it (M. Premoli).

https://doi.org/10.1016/j.ejor.2022.11.024
0377-2217/

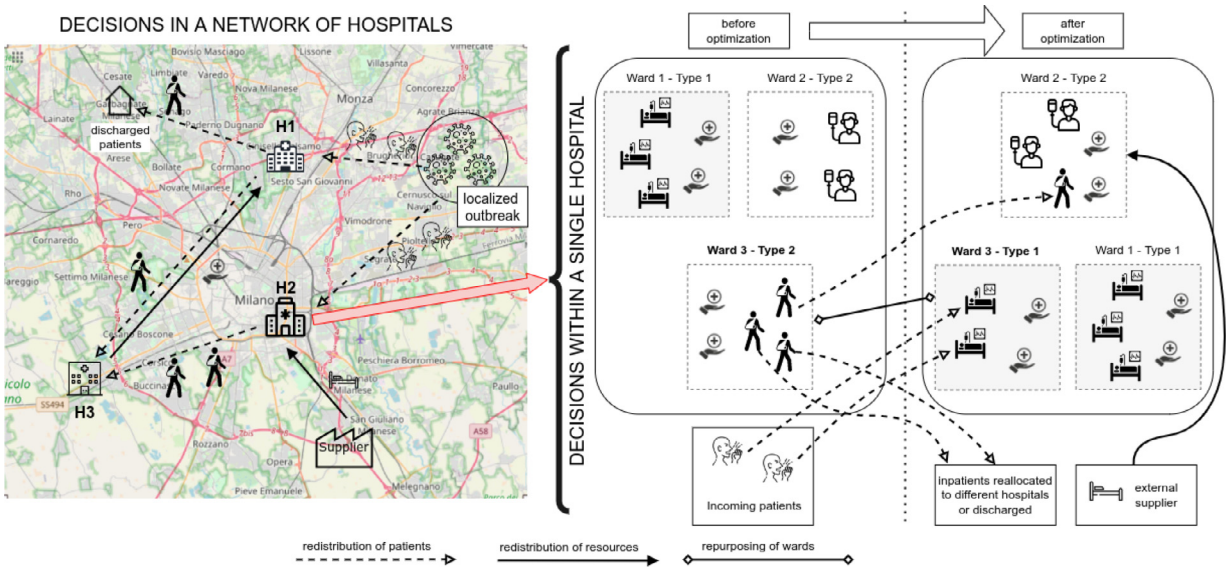


Fig. 1. Graphical example of the problem under study: a pandemic outbreak leads to a surge of incoming patients in a geographical area; to serve these latter, all hospitals of the area ($H1$, $H2$ and $H3$) need to redistribute both inpatients and resources among them; resources can be provided by external suppliers, inpatients can be discharged. Redistributions can happen also among wards of a single hospital, and a ward can be repurposed to host different type of patients.

depending on the intensity of the epidemic, and (b) reorganize the system in case one of the considered scenarios occurs.

To solve our problem we introduce a mathematical programming heuristic algorithm exploiting MILP decomposition methods and column generation algorithms. These are used to drive a very large-scale neighborhood search (VLSNS) procedure. The effectiveness of our MILP approach is evaluated from two perspectives. First, we assess its practical applicability to deal with problems arising from real-world situations. This includes the possibility of solving large-scale optimization problems populated with real data. Second, we consider its flexibility in prospective decision support tools. We consider various key performance indicators related both to the quality of service provided to the patients and to the economic effort arising from the implementation of the MILP solutions, adapting our algorithms to perform bi-objective optimization.

We perform scenario-based experiments, simulating the application of our model to the first wave of COVID-19 infections in Lombardy, a region of Northern Italy, during Spring 2020. The experimental results show that our methodology is effective both in terms of computational efficiency and in terms of solution quality.

Outline. This paper is organized as follows. In Section 2 we highlight similarities and novelties of our work with respect to the existing approaches for related variants of facility location-allocation problems. In Section 3 we provide our model, composed of a core of combinatorial features that may appear in generic rebalancing problems, and constraints arising from real-world emergency-specific features. In Section 4 we detail our mathematical programming heuristic and its embedding in a bi-objective optimization approach. To assess their computational effectiveness our algorithms are tested experimentally in Section 5, relying on a parametric analysis. We assess the value of our approach by comparing it to an adaptation of the exact algorithm of Corberán, Landete, Peiró, & Saldanha-da Gama (2020) to our problem. The conclusions of our research are given in Section 6.

2. Literature review

The literature on OR applications to epidemics logistics and large-scale emergencies is vast. Indeed, the scarcity of health resources is a critical issue triggering actions and reactions of different players. A full survey is beyond the scope of this section. We therefore refer the reader to Dasaklis, Pappis, & Rachaniotis (2012) and to Altay & Green (2006) for extensive literature reviews on these topics.

A first line of research that is strictly related with our work is the optimal location of facilities during large-scale emergencies (Boonmee, Arimura, & Asada, 2017). Problems of this type subsume the well-known *facility location-allocation* (FLA) problem, a combinatorial optimization problem where a set of facilities must be opened in either predetermined candidate sites (*discrete FLA*) or in a continuous region (*continuous FLA*) and customers must be allocated to the opened facilities so to cover the total customers' demand while minimizing the location and allocation costs. FLA problems naturally arise in business logistics (Klose & Drexl, 2005; Melo, Nickel, & Saldanha-Da-Gama, 2009). As such, classical FLA problems often overlook system congestion, facility unavailability and heterogeneity, resource scarcity and time-dependent demands, which instead characterize large-scale emergency applications, Jia, Ordóñez, & Dessouky (2007a).

FLA problems taking into account such emergency aspects are first introduced in Jia et al. (2007a) and Jia, Ordonez, & Dessouky (2007b). These two works adapt classical combinatorial optimization problems (such as the p-median, the p-center and the maximum covering problems, see e.g., Ahmadi-Javid, Seyedi, & Syam, 2017) and also stress the concept of quality of service in the modeling phase.

Two additional studies relevant to our discussion are presented in Ekici, Keskinocak, & Swann (2008) and Carr & Roberts (2010), where simulation models forecasting commodity demands are combined with algorithms to minimize the cost of locating emergency facilities needed to satisfy the demands. In both works, the optimization problem is modelled as an integer linear program (ILP) and is resolved iteratively over a rolling time horizon, updating dynamically the total commodity demand through the forecasting model. Ekici et al. (2008) considers the distribution of a single commodity through facilities belonging to a three-levels supply-chain and allows demand split, while Carr & Roberts (2010) considers a multi-commodity distribution without demand split but under resource-customer compatibility constraints. In our paper we combine several of these aspects: we model the presence of a global supplier and the redistribution of multiple health resources

(e.g., physicians, ventilators, beds) among the facilities, taking into account compatibility and demand split aspects.

Addressing the aspect of unavailability of facilities during emergencies, the research proposed in Huang, Kim, & Menezes (2010) develops a dynamic programming algorithm for solving, on a path network, a specific FLA problem based on a modified p-center problem and proposes a heuristic based on an ILP relaxation for solving the same problem on general networks. Although unavailability of facilities typically occurs in non-epidemic emergencies like earthquakes, this aspect is also partially present in our study: due to the high inter-human transmissibility of SARS-CoV-2, specific types of inpatients (e.g., geriatric ones) cannot be allocated to COVID-19 wards.

More recently, in Sun, DePuy, & Evans (2014) mono- and bi-objective mathematical models have been developed and solved exactly to tackle the problem of allocating patients and resources over a network of hospitals during an epidemic. The objectives of the models are the minimization of the total and the maximum distances travelled by patients for reaching the assigned hospitals. The most comprehensive model presented in Sun et al. (2014) considers several patient and resource types, along with patient-resource compatibility, resource shortages and external suppliers. Such high level of detail is also considered in our paper. Moreover, we will extend the above approach by also considering hospital ward repurposing and both resource and patient relocation over a network of hospitals.

Indeed, experts estimate these latter strategies to be effective in mitigating hospital congestion during large-scale epidemics, see e.g., Gagliano et al. (2020); Her (2020); Meschi et al. (2020); Scarfone et al. (2011). At the same time, some of them, as relocation of patients, have received little attention in the healthcare OR literature, see the discussion in Andersen, Nielsen, & Reinhardt (2017). In fact, in this context, OR-based approaches most often consider repurposing and reallocation within a single hospital (Andersen et al., 2017; Pishnamazzadeh, Sepehri, Panahi, & Moodi, 2021; Thomson, Nunez, Garfinkel, & Dean, 2009), while, to the best of our knowledge, they have not been applied to large networks of hospitals.

Two recent works exploit similar ideas for industrial FLA problems. The first of these problems is the *capacitated mobile facility location* (CMFL) introduced in Raghavan, Sahin, & Salman (2019). In the CMFL problem, a set of heterogeneous facilities of finite capacities must be relocated from their starting position to some destinations and must be assigned to a given set of customers so to satisfy their total demand without exceeding any facility capacity. The goal is to minimize the total distance covered by facilities (to reach the destination points) and customers (to reach the assigned facility). In Raghavan et al. (2019) the CMFL problem is modelled by means of a set partitioning ILP and solved by means of an effective branch-and-price algorithm. While relocation problems are common in the FLA literature (see e.g., Demaine et al., 2009; Melo, Nickel, & Da Gama, 2006 for two distinct perspectives) the CMFL presented in Raghavan et al. (2019) combines several characteristics that are also present in the problem studied in our paper: facilities (wards in our problem) are heterogeneous; the capacities of the facilities are finite; the facility relocation costs are not fixed, but depend on both origin and destination locations (initial and new ward types in our problem). There are relevant differences too: interpreting each patient type as a customer we allow demand splits, that is, patients of a given ward may be relocated to several new wards; in our problem ward capacities are not fixed, but depend on the assigned resources, and thus are part of the decision process; we consider a multi-commodity setting; finally, we take into account customer-resource compatibility. All these aspects, not modelled in Raghavan et al. (2019), allow a greater flexibility of our solutions in meeting the customers' demands; as a consequence, directly applying the methods of Raghavan et al. (2019) to our problem would yield, in general, infeasible or sub-optimal solutions.

In the context of FLA problems, the dependency of the ward capacities on the assigned resources is called *capacity transfer* and has been introduced in Corberán et al. (2020). The paper investigates the *facility location problem with capacity transfers* (FLPCT), a capacitated FLA problem where it is possible to increase the (physical or productive) capacity of a facility by transferring it from other facilities at some cost. In Corberán et al. (2020), the FLPCT is modelled by means of non-linear and linear integer programs, and solved using a branch-and-cut approach. The models proposed in Corberán et al. (2020) are suitable for homogeneous facilities (there is only one type of customer demand). We extend the capacity transfer feature to a multi-commodity setting which is additionally complicated by several constraints, arising from the specific real-world application.

In Table 1 we summarize the differences between the problem studied in this paper and the literature. For each relevant feature of our problem, we use the checkmark symbol (✓) if the feature is considered in the paper and we leave the entry blank otherwise.

3. Modeling relocation and reallocation in the health system

In the following we introduce our general modeling framework in terms of entities involved (Section 3.1), variables and constraints of our model (Sections 3.2–3.4) and optimization performance indicators (Section 3.5). The complete model is summarized in Appendix A, which also contains Tables reporting the complete sets of model entities (Table 4), variables (Table 5) and parameters (Table 6). An intuitive overview of our modeling choices is given in Fig. 2.

3.1. Model entities

The set of *hospitals* H (large rounded corner rectangles in Fig. 2) encompasses all hospitals that are available to relocate resources or patients. Each hospital $h \in H$ contains a set W_h of *logical wards* (circles within dashed-line rectangles in Fig. 2). Each logical ward is related to real-world *physical wards* in three (mutually exclusive) ways:

(a) in a 1-to-1 relationship: a logical ward represents the physical space of a single ward in a hospital;
(b) in a 1-to-many relationship: a logical ward represents a cluster of two or more physical wards, grouped to represent one or more specialties that, for the purpose of tactical optimization, may be pertinent to consider together. As an example, a logical ward of this type may represent a set of COVID-free wards hosting inpatients that cannot be discharged;
(c) in a many-to-1 relationship: a single physical ward is split in two or more logical wards, to manage their capacity at a smaller granularity. For example, if the structure of the rooms in a ward allows to be physically separated in smaller spaces with distinct access points, those become eligible as COVID-19 wards.

The mapping between physical and logical wards is performed by the decision maker in preprocessing, and is therefore part of data. Hence, in the following we simply use *ward* to refer to the logical ones. We consider $W = \bigcup_{h \in H} W_h$, and for each $w \in W$ we denote as $h(w)$ the element $h \in H$ such that $w \in W_h$, that is the hospital to which w belongs. In the set W we also include two special wards:

- the *homecare* ward $\bar{v}$, modeling domiciliary healthcare services such as telemedicine or assistance by qualified medical staff;

Table 1
Summary of literature gap. Column HRM represents our contribution. The first three lines are the essential combinatorial features of FLA problems. In our work 'customers' are patients; 'facility-customer compatibility' is checked even just on the basis of geographical distances and 'facility types' is checked when the type of facilities affects locations or allocations; our model takes into account additional real-world aspects, not listed in this table.

	HRM	Corberán et al. (2020)	Raghavan et al. (2019)	Sun et al. (2014)	Carr & Roberts (2010)	Ekici et al. (2008)
discrete facility location/relocation	✓	✓	✓		✓	✓
capacitated facilities	✓	✓	✓	✓	✓	✓
customer allocation	✓	✓	✓	✓	✓	✓
external supply of resources	✓	✓		✓	✓	✓
bi-objective model	✓			✓		
facility-customer compatibility	✓	✓		✓	✓	✓
multi-commodity resources	✓			✓		
facility types	✓		✓			
demand split	✓	✓		✓		✓
capacity transfer	✓	✓				

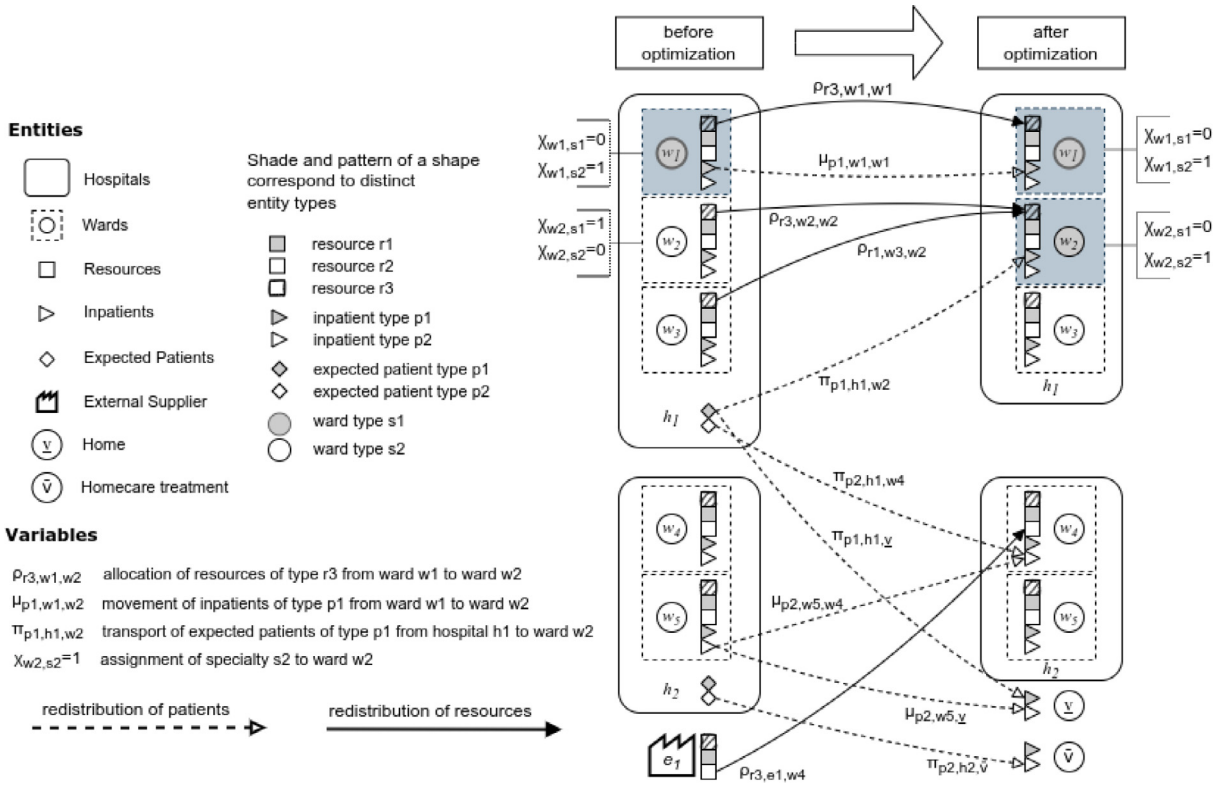


Fig. 2. An intuitive overview of our modeling framework. The left and right layers represent the settings before and after planning decisions, respectively, in a sample instance with $|H| = 2$ hospitals, $|V| = 5$ wards, $|R| = 3$ resource types and $|P| = 2$ patient types. A few of the decision options are represented as arrows between layers. Shade and patterns of a shape correspond to distinct entity types.

- the *home* ward $\underline{v}$ representing (as last resort) the discharging of patients from hospitals, with no domiciliary healthcare service except the standard one from the health system.

Wards and operators are characterized by a *specialty* chosen from a set S (different color shades of circles in Fig. 2), which contains all medical specialties appearing in the system, or additional aggregate specialties crafted by merging subsets of them when it is feasible for the purpose of planning to consider them as equivalent.

Patients are characterized by a *type* chosen from a set P (different color shades of triangles and diamonds in Fig. 2), that identifies both the ward specialty in which the patients need to be hosted, and the severity of their disease.

A set of *resources* R (squares in Fig. 2 with different patterns) contains one element for each type of human personnel, medical device, or consumable that is needed to treat inpatients. Clearly, material resource entities abstract from the extensive set of resources that are located in a hospital and R includes only that subset which more significantly limits the number of patients that can be hospitalized at once. As an example, the set R may contain elements for beds, ICU equipment, oxygen supply, etc. Human resource entities (i.e., physicians and nurses with their specialties) also belong to R: the corresponding elements are assumed to match logical wards. For example, COVID-related specialties (anaesthetists, pulmonologists and emergency physicians) and specialties related to non-deferrable pathologies (e.g., neurologists, cardiologists, etc.) should be considered as distinct resource categories, while other specialties may be merged in a single aggregate specialty to treat non-COVID and non-emergency patients.

Finally, a set of suppliers E (element e_1 in Fig. 2) models additional resources, which are external to the system of hospitals. These include for example additional device supplies, as well as

candidate locations to set up temporary support structures like field hospitals. To ease notation, we model suppliers as dummy wards, assuming $E \subset W$.

3.2. Decision variables

Our rebalancing model considers three main actions from the decision maker: (a) to change wards type, providing more room for expected peaks of demand, (b) to move resources from one ward to another and (c) to reassign patients between wards. In a tactical planning scenario, while actions (a) are crisp, actions (b) and (c) are meant more as estimates for future actions. These are described in the following. To ease their reading, decision variables and model parameters are summarized in Tables 5 and 6 of Appendix A.

Ward type. The first key assumption of our model is the following: the specialty of a ward can be changed, to meet an expected peak of inpatients demand which is far from that of standard load. Such a change requires suitable setup time, and often different resources to be made available in the ward.

We therefore introduce for each ward $w \in W$ and each specialty $s \in S$ a binary variable $\chi_{w,s}$, taking value 1 if ward w is assigned to specialty s, 0 otherwise. In the example of Fig. 2, ward w_1 keeps its specialty ($\chi_{w_1,s_1} = 0$, $\chi_{w_1,s_2} = 1$) while ward w_2 changes it ($\chi_{w_2,s_1} = 0$, $\chi_{w_2,s_2} = 1$).

Resources allocation. The second key assumption of our model is the following: during emergency situations, staffers, device and material resources can be temporarily moved from one ward to another, potentially in different hospitals.

Accordingly, we introduce for each resource type $r \in R$, each source ward $w_1 \in W$ and each destination ward $w_2 \in W$ a variable $\rho_{r,w_1,w_2} \in \mathbb{R}_{\geq 0}$, representing the amount of resources of type r that need to be moved from w_1 to w_2. Since $E \subset W$, these variables describe also the amount of resources transferred from the external suppliers to wards. In the example of Fig. 2, a certain amount ρ_{r_1,w_3,w_2} of resources of type r_1 is moved from ward w_3 to ward w_2, another amount ρ_{r_1,w_1,w_1} of resources r_1 is kept in ward w_1 and ρ_{r,e_1,w_2} represents the amount of resource r obtained by w_2 from supplier e_1.

Patients' allocation. Under normal load, the health system expects patients to refer to the nearest hospital. However, under system stress conditions, handling all patients of a certain type in their nearest hospital might lead to the collapse of certain wards. Therefore, assuming to have a forecasting on the expected number of patients of each type, we plan in advance the (expected) number of new patients of each type $p \in P$ appearing in each hospital $h \in H$ to be transported for treatment to a ward $w \in W$, either in the same or in another hospital. This is modeled by introducing a set of variables $\pi_{p,h,w} \in \mathbb{R}_{\geq 0}$.

Furthermore, the third key assumption of our relocation mechanism is the following: even existing inpatients can be moved from one ward to another, potentially located in a different hospital, provided the destination ward is eligible to host the patient. We also assume that the treatment of some types of patients can be postponed, and even that inpatients of selected types of diseases can be discharged from the hospital and treated by means of a dedicated domiciliary healthcare system. That practice is in fact the main one experimented with success to reduce wards congestion during the COVID-19 emergency (Zuccotti, Bertoli, Foppiani, Verduci, & Battezzati, 2020).

We therefore include in the model also a set of variables $\mu_{p,w_1,w_2} \in \mathbb{R}_{\geq 0}$ that represent the number of inpatients of type $p \in P$ to move from ward $w_1 \in W$ to ward $w_2 \in W$. Wards w_1 and w_2 can belong to the same hospital, or different ones; w_2 can even be the special ward representing the domiciliary healthcare system or the discharge of the patient.

For instance, in Fig. 2 a certain number μ_{p_1,w_1,w_1} of inpatients of type p_1 is kept in ward w_1, while a number $\mu_{p_2,w_5,\underline{v}}$ is discharged from hospitals. At the same time, a number π_{p_1,h_1,w_2} of new patients of type p_1 is expected to be transported from hospital h_1 to ward w_2, a number π_{p_2,h_1,w_4} to ward w_4 (which is located in hospital h_2), and a number $\pi_{p_2,h_2,\bar{v}}$ to dedicated homecare services.

3.3. Modeling system rebalancing

In order to detail a valid tactical plan, the setting of variables must respect the following conditions.

Ward types single assignment and patients compatibility. A single specialty is assigned to each ward $w \in W$, chosen among a subset of specialties which are considered to be valid for w depending on its functional and physical structure. Let $a_{w,s}$ be a coefficient, set to 1 if it is feasible for ward $w \in W$ to have type $s \in S$, 0 otherwise:

$$\sum_{s \in S} a_{w,s} \chi_{w,s} = 1 \quad \forall w \in W \tag{1}$$

Patients and resources flow consistency. Patients and resources are modeled as units of flow, originating in source wards and sent to destination wards. Let $q^0_{p,w}$ be the number of existing inpatients of type p in ward w and $q_{p,h}$ be the number of new patients of type p expected to arrive for hospitalization in hospital $h \in H$.

It is not always possible to move a patient from one ward to another. Typical restriction are due to travel distance or time. For each patient type $p \in P$ and hospitals $h_1, h_2 \in H$, let coefficient a_{p,h_1,h_2} take value 1 if moving patients of type p from h_1 to h_2 is possible, 0 otherwise.

The number of patients moving from each ward must match these initial values, i.e., all existing inpatients in wards must be moved to another compatible ward, and the treatment of all new patients must be planned by moving them to compatible wards:

$$\sum_{w_2 \in W} a_{p,h(w_1),h(w_2)} \mu_{p,w_1,w_2} \geq q^0_{p,w_1} \qquad \forall p \in P, w_1 \in W \tag{2}$$

$$\sum_{w \in W} a_{s,h,h(w)} \pi_{p,h,w} \geq q_{p,h} \qquad \forall p \in P, h \in H \tag{3}$$

We remark that in (2) w_1 can be equal to w_2, in which case the inpatients do not move. Similarly, in (3) $h(w)$ can be equal to h, in which case the new patients are treated directly in the hospital where they appear. These latter actions are always possible, hence the corresponding coefficients 'a' are equal to 1; moreover, variables μ and ρ linked to coefficients 'a' of value 0 are set to 0 in preprocessing. The transportation of resources among wards, including suppliers, must satisfy a symmetric condition: no more resources than the available ones can be moved from the initial wards.Letting $q^0_{r,w}$ be the amount of resource $r \in R$ initially available in ward $w \in W$, we get constraints:

$$\sum_{w_2 \in W} \rho_{r,w_1,w_2} \leq q^0_{r,w_1} \qquad \forall r \in R, w_1 \in W \tag{4}$$

Wards capacity. Wards have capacities: they can host only a limited amount of each resource. At the same time, wards can host only a limited number of inpatients for a specific disease, depending on both their structure and the amount of allocated resources.

For each resource $r \in R$ and for each ward $w \in W$, let $m_{r,w}$ be the maximum amount of resource r that the ward w can host. The amount of resources sent to each ward $w_2 \in W$ cannot exceed its capacity:

$$\sum_{w_1 \in W} \rho_{r,w_1,w_2} \leq m_{r,w_2} \qquad \forall w_2 \in W, r \in R \tag{5}$$

Similarly, let $n_{p,r}$ be the amount of resource r required by each unit of patient of type $p \in P$. The amount of each resource $r \in R$

required by all patients sent to each ward $w_2 \in W$ cannot exceed the availability of r in w_2:

$$\sum_{p\in P} n_{p,r}\left(\sum_{h\in H}\pi_{p,h,w_2} + \sum_{w_1\in W}\mu_{p,w_1,w_2}\right) \le \sum_{w_1\in W}\rho_{r,w_1,w_2} \quad \forall w_2 \in W\backslash\{\bar{v},\underline{v}\}, r \in R \tag{6}$$

For instance, in Fig. 2 sending $\rho_{r_1,w_2,w_2} + \rho_{r_1,w_3,w_2}$ units of resource r_1 to w_2 allows to host π_{p_2,h_1,w_2} new patients of type p_2 in ward w_2. Similarly, acquiring ρ_{r_3,e_1,w_4} units of resource from supplier e_1 allows to host π_{p_2,h_1,w_4} new patients in w_4. Resources are not consumed in the case of transportation to wards '$\bar{v}$' and '$\underline{v}$', representing homecare treatment and patient's home, respectively (e.g. for discharged inpatients or postponed elective treatments).

Wards compatibility. Wards can host only specific types of patients. For each patient type $p \in P$ and for each specialty $s \in S$, let $a_{p,s}$ be a binary parameter, set to 1 if patients of type p can be hosted in wards of specialty s, 0 otherwise. Let also

$$\bar{q}^0_{p,w_1,w_2} = \min\{q^0_{p,w_1}, \min_{r\in R}\lfloor m_{r,w_2}/n_{p,r}\rfloor\}$$

be an upper bound on the number of existing inpatients of type p that can be moved from w_1 to w_2, and let

$$\bar{q}_{p,h,w} = \min\{q_{h,p}, \min_{r\in R}\lfloor m_{r,w}/n_{p,r}\rfloor\}$$

be an upper bound on the number of new patients of type p that appear at hospital h and are sent to ward w. A ward cannot host new patients of a certain type unless it is switched to a compatible speciality:

$$\pi_{p,h,w} \le \bar{q}_{p,h,w}\sum_{s\in S} a_{p,s}\chi_{w,s} \qquad \forall p \in P, \forall h \in H, \forall w \in W. \tag{7}$$

For instance, in Fig. 2, ward w_2 must change its specialty to s_2 ($\chi_{w_2,s_2} = 1$) to host patients of type p_1 (π_{p_1,h_1,w_2}). Similarly, a ward cannot host existing inpatients of a certain type unless its speciality is compatible:

$$\mu_{p,w_1,w_2} \le \bar{q}^0_{p,w_1,w_2}\sum_{s\in S} a_{p,s}\chi_{w_2,s} \quad \forall p \in P, \forall w_1 \in W, \forall w_2 \in W. \tag{8}$$

3.4. Modeling emergency-specific features

In the following we introduce additional constraints arising from the COVID-19 emergency in Northern Italian health system, as suggested by domain experts (e.g. Villani, 2020).

Keeping critical inpatients at their hospital. Not every inpatient can safely leave his/her hospital to be transported to other ones. We assume that a fraction $f_{p,w} \in [0, 1]$ of patients of each type $p \in P$ that are initially placed in ward $w \in W$ cannot leave the hospital hosting them. Accordingly, the following set of constraints is added to the model:

$$\sum_{w_1\in W_{h(w)}} \mu_{p,w,w_1} \ge q^0_{p,w} f_{p,w} \quad \forall p \in P, w \in W \tag{9}$$

Providing support resources for comorbidities. The treatment of each patient may require more than a single specialist. It is certainly the case of COVID-19, where a large part of critical inpatients suffers other chronic diseases. As discussed in the previous subsection, these inpatients require a set of main mandatory resources of COVID-19 wards, in exclusive way. Additionally, they need a set of support resources only in case of need, or as low frequency routine. These latter can be provided on a *proximity* basis: inpatients of a certain ward are serviced by resources hosted in potentially different wards, if they are located within a given distance range.

Accordingly, we assume that the amount $\hat{n}_{p,r}$ of support resources of type r needed by patients of type p is given, as well as the maximum distance $\hat{d}$ between the patients and the ward hosting the support resources. Introducing variables τ_{r,w_2,w_3} describing the amount of resource r located in w_3 and serving inpatients in w_2, our model is enriched by the following set of constraints:

$$\sum_{w_3: d_{h(w_2),h(w_3)}\le\hat{d}} \tau_{r,w_2,w_3} \ge \hat{n}_{p,r}\Big(\sum_{h\in H}\pi_{p,h,w_2} + \sum_{w_1\in W}\mu_{p,w_1,w_2}\Big) \quad \forall p \in P, r \in R, w_2 \in W\backslash\{\underline{v}\} \tag{10}$$

$$\sum_{w_2\in W}\tau_{r,w_2,w_3} \le \sum_{w_1\in W}\rho_{r,w_1,w_3} \qquad \forall w_3 \in W, r \in R \tag{11}$$

Constraints (10) are imposed also for patients moved to 'homecare' treatment (ward $\bar{v}$); indeed, we model the homecare treatment as a non exclusive use of resources. Finally, we remark that the possibility of patients to change type *during* the planning horizon is not taken into account, since we assume such an option to be important at an operational level, but to have a limited impact in tactical planning, for which our model is designed.

Alternative resources. Another mean of coping with a critical lack of resources during emergencies is to replace specific ones with viable alternatives. A relevant case is given by the choice of physicians in COVID-19 sub-intensive care units: wards formally require either anaesthetist, pulmonologist or emergency physicians. However, different specialists are allowed to support, if at least some among them hold the specific ICU skills.

Accordingly, let $r^A \in R^A$ be a set of resources which are considered to be equivalent, and let $\bar{n}_{p,r^A}$ be the amount of alternative resources in the set r^A needed by each patient of type p. We impose

$$\sum_{p\in P}\bar{n}_{p,r^A}\Big(\sum_{h\in H}\pi_{p,h,w_2} + \sum_{w_1\in W}\mu_{p,w_1,w_2}\Big) \le \sum_{w_1\in W, r\in r^A}\rho_{r,w_1,w_2} \quad \forall w_2 \in W\backslash\{\bar{v},\underline{v}\}, r^A \in R^A \tag{12}$$

From an application point of view, introducing alternative resources (and therefore constraints (12)) allows to reduce the set of specific ones required by some types of patients, thereby relaxing some constraints in the set (6).

3.5. Key performance indicators

There are many aspects that need to be balanced to obtain an effective planning. In the following we detail each of them as a Key Performance Indicator (KPI). Then we discuss on how it is more appropriate to manage them, i.e., imposing target constraints or combining them into an objective function. We keep a minimization philosophy: for each KPI, the lower the better.

Quality of Service (QoS). The QoS in our system is measured regarding the patients' side. In particular, we measure (a) the number of inpatients that are moved from one hospital to another and (b) the number of patients that are not treated in a hospital (i.e., those either discharged or for which the hospitalization has been delayed even if required).

Let $c_{p,w_1,w_2} \in \mathbb{R}_{\ge 0}$ be part of data, representing the cost for moving inpatients of type p from ward w_1 to ward w_2; similarly, let $c_{p,h,w} \in \mathbb{R}_{\ge 0}$ be the cost for relocating incoming patients of type p from hospital h to ward w. We define the following QoS measure:

$$C^{QoS} = \sum_{p\in P, w\in W}\left(\sum_{h\in H} c_{p,h,w}\pi_{p,h,w} + \sum_{w_1\in W} c_{p,w_1,w}\mu_{p,w_1,w}\right) \tag{13}$$

We remark that sending patients to homecare (ward $\bar{v}$), as well as resorting to their discharge from the hospitals without dedicated domiciliary healthcare (ward $\underline{v}$) imply costs that affect the QoS. The

latter is in fact assumed to be set by a decision maker to very high values. A high level of service is associated with low KPI values.

Logistics. As a second KPI we take into account the economic effort to setup wards and transport resources. Let $s^0(w)$ be the element $s \in S$ that identifies the initial speciality of ward w and $c_{s_1,s_2} \in \mathbb{R}_{\geq 0}$ be the cost to change the specialty of a ward from s_1 to s_2. Moreover let $c_{r,w_1,w_2} \in \mathbb{R}_{\geq 0}$ be the cost to move resource of type r from ward w_1 to ward w_2. We compute the economic effort as:

$$C^E = \sum_{w \in W, s_2 \in S} c_{s^0(w),s_2} \chi_{w,s_2} + \sum_{r \in R, w_1,w_2 \in W} c_{r,w_1,w_2} \mu_{r,w_1,w_2} \tag{14}$$

The lower this KPI, the better. We remark that the arrangement of temporary field hospitals has a cost, that can conveniently be encoded through coefficients c_{s_1,s_2}.

Reallocation effort. As a third KPI we consider the time needed to setup wards and move resources.

Let t_{r,w_1,w_2} (resp. t_{p,w_1,w_2}) be the time needed to move one unit of resource $r \in R$ (resp. patient of type $p \in P$) from ward $w_1 \in W$ to ward $w_2 \in W$; let also t_{s_1,s_2} be the time needed to convert a ward from specialty $s_1 \in S$ to $s_2 \in S$. We define

$$T^R = \sum_{w \in W, s_2 \in S} t_{s^0(w),s_2} \chi_{w,s_2} + \sum_{w_1,w_2 \in W} \left(\sum_{r \in R} t_{r,w_1,w_2} \rho_{r,w_1,w_2} + \sum_{p \in P} t_{p,w_1,w_2} \mu_{p,w_1,w_2} \right)$$

representing the number of working hours needed to perform the full relocation. We remark that the model does not include the scheduling of these resources: they only represent the overall effort required. The actual time to perform reallocation depends therefore on the amount of personnel actually available.

Expected patients relocation time. Part of the flexibility of the model in managing the capacity is obtained by assuming new patients to be moved from the nearest hospital to other ones. The expected moving time of these patients is also an important KPI, being part of the overall quality of service. Let $t_{p,h,w}$ be the time needed to move an incoming patient of type $p \in P$ from hospital h to ward w. We define

$$T^A = \frac{\sum_{p \in P, h \in H, w \in W} t_{p,h,w} \pi_{p,h,w}}{\sum_{p \in P, h \in H} q_{p,h}}$$

as the expected moving time of a single new patient.

Combining KPIs. There are different ways in which these KPIs can be combined in a pertinent way. In the following we assume fixing bounds on time KPIs to be more relevant for a decision maker than minimizing them. We therefore assume that the decision maker is fixing the value of two parameters $\bar{T}^R$ and $\bar{T}^A$, representing the available man hours limit of time to move initial resources and inpatients, and the maximum allowed average time for the displacement of incoming patients, respectively. We impose

$$T^R \leq \bar{T}^R \tag{15}$$

$$T^A \leq \bar{T}^A \tag{16}$$

and focus on the optimization of the cost KPIs. The general model we start from is:

$$\begin{aligned} \text{minimize } & \alpha C^{QoS} + \beta C^E \\ \text{s.t. } & (1)\text{–}(16) \\ & C^{QoS} \leq \bar{C}^{QoS} \\ & C^E \leq \bar{C}^E \\ & \chi_{w,s} \in \{0,1\} \quad w \in W \text{ and } s \in S \\ & \pi_{p,h,s},\ \rho_{r,w,w'},\ \mu_{p,w,w'},\ \tau_{r,w',w''} \in \mathbb{R}_{\geq 0} \\ & \quad p \in P, h \in H, r \in R, w, w', w'' \in W \end{aligned} \tag{17}$$

where α and β are two real nonnegative parameters, and $\bar{C}^{QoS}$ and $\bar{C}^E$ are upper bounds for C^{QoS} and C^E, respectively. We will consider two specializations of (17): in the first one $\bar{C}^{QoS} = \bar{C}^E = +\infty$ and $\alpha, \beta > 0$, so that we minimize a weighted sum of unbounded cost KPIs; in the second specialization we adopt a multi-objective approach setting to 0 precisely one weight α or β while imposing finite upper bounds on the related KPIs. The above problem (17) will be referred to as *hospital resource management* (HRM).

4. Optimization algorithms

In a preliminary phase, we experimented on solving model (17) with the commercial solver Gurobi 9.5 (Gurobi Optimization (2021)). Due to the size of real-world instances, even the resolution of the LP relaxation (root node) ran out of memory on a workstation equipped with 32 gigabyte of RAM.

In fact, model (17) is meant more as a baseline for the design of a mathematical programming heuristic than as a direct resolution tool. In our algorithms we search for good solutions through a VLSNS approach. The main idea is to iteratively (a) choose a subset of binary variables, fixing them to promising values, and (b) explore the space of all possible solution completions. Neither step (a) nor step (b) is a trivial task. To accomplish step (a) we design a column generation (CG) algorithm, exploiting the structure of reduced costs arising during its execution. To accomplish step (b) we solve restricted MIPs by a truncated run of a general purpose solver. The process iterates in a local search fashion.

In this context, the CG algorithm has several advantages over e.g., the direct resolution of the LP relaxation of formulation (17): first, it lets us run the VLSNS procedure repeatedly, thus exploring a larger set of feasible solutions; moreover, it exploits a natural decomposition of model (17), thus limiting the computational burden; finally, the valid lower bound it provides lets us control consistently the quality of the heuristic solutions generated in the process.

We start by describing our column generation algorithm (Section 4.1), then we proceed by describing the VLSNS procedure (Section 4.2).

4.1. Column-generation algorithm

Decomposition scheme. We employ the well-known scheme of Dantzig–Wolfe relaxation to get a valid lower bound to formulation (17). For a general treatment of this approach we refer the reader to Desrosiers & Lübbecke (2005). The details of our reformulation, and its full notation, are reported in Appendix B.

The overall algorithm works as follows. We relax (17) by mapping constraints (2), (3), (4), (10), (15) and (16) as constraints of an extended formulation called *master problem*, whose number of columns grows combinatorially, encoding the extreme points of the convex hull of the remaining constraints. Then, we solve this extended formulation by means of column generation: we start with a small subset of columns, solve this restricted master problem (RMP), and use the values of dual variables to find which columns are left out, having minimum reduced cost (*pricing problem*).

Our choice of relaxed constraints has three appealing features. First, it allows the pricing problem to disaggregate in one independent subproblem for each $w \in W$. Each of them contains the set of binary variables $\chi_{w,s}$ corresponding to a particular value of $w \in W$, as well as other sets of continuous variables, also related to a particular $w \in W$. Therefore, these $|W|$ pricing subproblems can be solved independently. Second, pricing subproblems do not possess the integrality property, which means that the lower bound obtained with our relaxation is potentially stronger than that given by the continuous relaxation of (17). Third, we are able to exploit

the multiple-choice structure of constraints (1), to solve them efficiently: we optimize each pricing problem $w \in W$ by iteratively fixing one of the $\chi_{w,s}$ variables to 1 and the others to 0, solving the remaining LP, and retaining as final pricing solution for each $w \in W$ the best one among these $|S|$ iterations. This yields optimal pricing solutions in polynomial time.

If no column of negative reduced cost is found by pricing, then the optimal RMP solution is optimal for the full master problem as well, and therefore represents a valid dual bound to (17). Otherwise, the RMP is enriched by the negative reduced cost columns produced by pricing and the process is iterated until convergence (Desrosiers & Lübbecke, 2005).

Constraints strengthening and handling We strengthen the pricing sub-problem associated with $w' \in W$, by including in its formulation the following constraint:

$$\rho_{r,w_1,w'} \leq q^0_{r,w_1}, \ \forall r \in R, w_1 \in W \tag{18}$$

The above is a disaggregation of constraints (4). Its meaning is that the amount of resource $r \in R$ sent from $w_1 \in W$ to $w' \in W$ cannot exceed the total available amount q^0_{r,w_1} of r in w_1.

In preliminary experiments we observed numerical issues and very slow convergence of our column generation algorithm, that we were able to ascribe to the presence of the proximity resource constraints (10), which appear in the master problem as 10-DWR, see Appendix B.

Therefore, we treat such constraints in a lazy way in an iterative process. Initially, we remove them from the master problem. When our CG algorithm ends, we check if the best heuristic solution found violates (10). The violation check is performed by solving a linear program, described in Appendix C. If one or more constraints of family (10) are violated, the corresponding constraints 10-DWR are added to the master problem and a further iteration of resolution of the CG algorithm is performed, using the updated master problem, and setting the previous master solution as a warm start.

In our experiments this technique provided substantial speedups: the process always ended at the first iteration, as we found that it was always possible to satisfy constraints (10) in the best heuristic solution found at the end of the first CG procedure by means of our LP postprocessing model. Intuitively, variables τ are used only to enforce consistency of constraints (10) and (11), and do not appear in the objective function. Constraints 10-DWR are however made easy to satisfy in the master, since the structure of constraints (6) and (12), which are additionally handled in a convexified way in the pricing problems, are pushing their left and right hand sides apart.

Column generation speedup. We designed the following stopping criterion for the column generation algorithm: letting $\bar{c}_w$ be the objective function value of the pricing sub-problem (equation (20) in Appendix B) for $w \in W$, we stop the column-generation algorithm (a) if no negative $\bar{c}_w$ is found in the iteration, or (b) after 500 iterations or (c) as soon as the gap between the objective function value $\bar{z}$ of the RMP and the valid lower-bound $\bar{z} + \sum_{w \in W, : \bar{c}_w < 0} \bar{c}_w$ is less than 1% (the validity of this lower-bound is shown e.g., in Desrosiers & Lübbecke (2005, p. 11)).

We further speed up our column-generation algorithms by stopping each pricing sub-problem after 2 seconds or after 1000 simplex iterations. To limit the number of columns added to the RMP after solving the pricing sub-problems, their optimal solutions are sorted by non-decreasing reduced cost. Following this order, at most $\lfloor |W|/2 \rfloor$ columns with negative reduced cost are added at each iteration.

4.2. Very large-scale neighborhood search.

Model (17) is not tractable for direct optimization. Its combinatorial complexity derives from the choice of χ variables. In fact, once these are fixed, a linear program remains, which can additionally be split in several subproblems. Clearly, the choice for optimal χ values is highly non-trivial. To effectively optimize it, we devised the following algorithm.

An incumbent solution χ^0 is initially generated by fixing a subset F of χ variables. Our initialization policy is detailed in Section 5. We then iteratively improve χ^0 along the execution of our column-generation algorithm by exploiting the information contained in the optimal solution of the pricing sub-problems.

More precisely, let $S_F := \{s \in S : \chi_{w,s} \in F \text{ for some } w \in W\}$; at a given iteration of the column-generation algorithm and for every $w \in W$ and $s \in S$, let $\bar{\chi}_{w,s}$ be the value of variable $\chi_{w,s}$ in the best heuristic solution generated so far and let $\bar{c}_{w,s}$ be the value of (20) fixing $\chi_{w,s} = 1$. Note that $\bar{c}_{w,s}$ is the reduced cost of the associated column. We define for each $\bar{s} \in S_F$:

$$\Delta_{\bar{s}} = \sum_{w \in W} \bar{c}_{w,\bar{s}} - \sum_{w \in W : \bar{\chi}_{w,\bar{s}} = 1} \bar{c}_{w,\bar{s}} = \sum_{s \in S : \bar{\chi}_{w,\bar{s}} = 0} \bar{c}_{w,\bar{s}}. \tag{19}$$

Interpreting the reduced cost of a variable as the potential improvement in the objective function yielded by increasing that variable of one unit, the smaller the value of $\Delta_{\bar{w}}$, the more likely the ward type $\bar{w}$ is assigned to the correct wards $w \in W$ by the vector $\bar{\chi}$.

This suggests to sort the ward types $s \in S_F$ by non-decreasing values of Δ_s and to define $S' \subseteq S_F$ as the set of the first $|S_F|$-UNFIXED_WARD_TYPES ward types in the ordering, where UNFIXED_WARD_TYPES is a fixed parameter (the actual value used in our experiments is specified in Section 5). Then, letting $F' = \{\chi_{w,s} : w \in W, s \in S'\}$ we solve to optimality model (17) after fixing to value $\chi^0_{w,s}$ each variable $\chi_{w,s} \in F'$, thus obtaining a new feasible solution. Note that, by definition, $F' \subseteq F$, hence a smaller number of variables is fixed with respect to the initial heuristic solution. This implies that all heuristic solutions generated during the execution of the column-generation algorithm are not worse than the initial solution. At each step, we keep the best generated heuristic solution.

Our complete mathematical programming heuristic (MPH for short) is summarized as pseudo-code in Algorithm 1. In a generic setting, conditions can be imposed for a run of VLSNS (line 12 of Algorithm 1). In our implementation, we define three conditions: (1) the value *LB* computed at line 7 of Algorithm 1 must be greater than zero; (2) the lower bound computed in the current CG iteration LB' must improve the best lower bound *LB* found so far and (3) is the first use of set F'. Indeed, in the process described above, it is possible that the same subset F' is generated in multiple column-generation iterations.

4.3. Multi-objective optimization

As detailed in Section 3.5, our problem includes two objectives: the economical costs C^E and the penalties of quality of service C^{QoS}. We have designed an approach to explore the Pareto set of solutions of HRM. The literature on multi-objective optimization is rich (Marler & Arora, 2004) and includes various approach such as fuzzy compromise programming (Parra, Terol, Gladish, & Uría, 2005) and augmented ϵ-constraint (Mavrotas, 2009). Here, our main aim is to get a better understanding of the relationship between the two terms of the objective function. Our approach is inspired by the augmented ϵ-constraint method, and its pseudo-code is presented in Algorithm 2.

The two terms C^E and C^{QoS} affect the model via the value of their weight in the objective function (resp. β and α) and the threshold value (resp. $\bar{C}^E$ and $\bar{C}^{QoS}$). First, the *utopia* values of the two terms are computed (resp. UB^U_E and UB^U_{QoS}): our algorithm MPH is run twice, setting the weight of the term not to consider

Algorithm 1: Mathematical programming heuristic (MPH) for HRM.

Input : an instance of HRM, a value of UNFIXED_WARD_TYPES, a set $F \subseteq \{\chi_{w,s} : w \in W, s \in S\}$, an starting feasible solution χ^0

Output: a valid lower bound LB, a primal solution $\bar{\chi}$ to the instance of HRM and its value $UB^\star$

$S_F := \{s \in S : \chi_{w,s} \in F \text{ for some } w \in W\}$;
$UB^0, \bar{\chi} =$ value and solution to HRM instance after fixing variables in F; $UB^\star = UB^0$;
initialize RMP with columns from $\bar{\chi}$;
repeat
  $\bar{z}, \bar{\theta}$ = value and solution to RMP;
  $\bar{c}_w = \{\min_{s \in S}\{\bar{c}_{w,s} = \text{solve (20) with } \theta \text{ and } \chi_{w,s} = 1\}, \forall w \in W\}$;
  $LB' = \bar{z} + \sum_{w \in W : \bar{c}_w < 0} \bar{c}_w$; $LB = \max\{LB', LB\}$;
  compute $\Delta_{\bar{s}}$ using $\bar{c}_{w,s}$ with (19);
  sort S_F by non-decreasing value of $\Delta_{\bar{s}}$;
  $S' =$ first $|S_F| -$ UNFIXED_WARD_TYPES elements in the ordering;
  $F' = \{\chi_{w,s} : w \in W, s \in S'\}$;
  if *conditions to run VLSNS* **then**
    UB, χ = value and solution of HRM instance after fixing variables in F' to value χ^0;
    if $UB < UB^\star$ **then**
      $UB^\star = UB$, $\bar{\chi} = \chi$;
    end
  end
  add columns with negative $\bar{c}_w$ to RMP;
until *CG stopping criterion*;

Algorithm 2: Multi-objective optimization scheme for HRM.

Input : an instance of HRM, a value of UNFIXED_WARD_TYPES, a set $F \subseteq \{\chi_{w,s} : w \in W, s \in S\}$, a value of χ^0, a value of GRID_LENGTH

Output: a valid lower bound LB, a primal solution χ to the instance of HRM and its value UB for the $2 \cdot$ GRID_LENGTH combinations of augmented ϵ-constraints setting and utopia values for C^E and C^{QoS}

/* unless otherwise stated $\bar{C}^{QoS} = +\infty$ and $\bar{C}^E = +\infty$ */
/* utopia for C^E */
$UB^U_E, \chi^U_E, LB^U_E \leftarrow$ MPH with $\alpha = 0, \beta = 1$;
/* utopia for C^{QoS} */
$UB^U_{QoS}, \chi^U_{QoS}, LB^U_{QoS} \leftarrow$ MPH with $\alpha = 1, \beta = 0$;
/* solution with equally weighted C^E and C^{QoS} */
$UB^\star, \chi^\star, LB^\star \leftarrow$ MPH with $\alpha = 1, \beta = 1$;
compute $C^E_\star$ and $C^{QoS}_\star$ from $UB^\star$;
/* intermediate values between utopias and solution $UB^\star$ */
for $i = 0..$GRID_LENGTH **do**
  /* if $i > 0$, initialize RMP of MPH with all columns found up to iteration $i-1$ */
  $C^E_{\epsilon\text{-i}}, \chi^{\epsilon\text{-i}}_E, LB^{\epsilon\text{-i}}_E \leftarrow$ MPH with $\alpha = 0, \beta = 1, \bar{C}^{QoS} = UB^U_{QoS} + i \cdot \frac{C^{QoS}_\star - UB^U_{QoS}}{\text{GRID\_LENGTH}+1}$
  $C^{QoS}_{\epsilon\text{-i}}, \chi^{\epsilon\text{-i}}_{QoS}, LB^{\epsilon\text{-i}}_{QoS} \leftarrow$ MPH with $\alpha = 1, \beta = 0, \bar{C}^E = UB^U_E + i \cdot \frac{C^E_\star - UB^U_E}{\text{GRID\_LENGTH}+1}$
end

to value 0, and not imposing any threshold value (or, equivalently, setting $\bar{C}^{QoS} = \bar{C}^E = +\infty$)

To compute the *nadir* values, we run MPH setting the weights with same value 1 and no threshold value imposed. This execution provides a heuristic solution $\chi^\star$ with value $UB^\star$. This value is decomposed in the two terms $C^E_\star$ and $C^{QoS}_\star$, which are used as nadir values.

The gap between utopia and nadir is explored setting a number of equally spaced grid points between them (parameter GRID_LENGTH in Algorithm 2); each grid point identifies the threshold value of a term while optimizing the other term. For example, the optimization of C^E in the first grid point is carried running MPH with $\alpha = 0$ and $\beta = 1$ and imposing a threshold value $\bar{C}^{QoS} = UB^U_{QoS} + (C^{QoS}_\star - UB^U_{QoS})/(\text{GRID\_LENGTH} + 1)$. On the ith grid point $\epsilon - i$, our algorithm MPH provides a heuristic solution $\chi^{\epsilon-i}$, its value $C^E_{\epsilon-i}$ and a valid lower bound $LB^{\epsilon-i}_E$.

We exploit the CG framework of our algorithm MPH to speed up the computation of grid points. We start the computation from the grid point with the stricter threshold value, loosening this value in each subsequent visited grid point. At the end of the computation of the grid point i, the computation of the following point $i+1$ is initialized including to the RMP all columns generated up to iteration i, which are feasible also for point $i+1$.

5. Experimental evaluation

In this section we test on the effective use of our algorithms for tactical response to an epidemic in a vast geographical area. To this end, we define instances of realistic size starting from data available at public institutional repositories. These include: the number and location of hospitals and their wards in Lombardy region; staff size and historical number of inpatients for each ward; number of incoming COVID-19 patients during the considered time period.

The data collection was not easy, as their sources are heterogeneous and some of them are unformatted plaintext reports. The precise repositories URLs, the formulas and the assumptions used to setting up parameter values are found in a companion technical report available in Premoli, Barbato, & Ceselli (2021), together with the complete instances used in our experiments. While most of parameters are fixed to specific values, we perform a scenario-based analysis by varying some of the parameters of our model. In the following we summarize only the main aspects of the parameters initialization.

Parameters initialization. Our model considers 86 hospitals in set H and 486 logical wards in set W. These latter include home and homecare wards ($\underline{v}, \bar{v}$), one global supplier and 483 hospital wards (retrieved from repository in Regione Lombardia (2020c), listing hospitals and wards in Lombardy as of September 2020). The costs of transporting resources and patients between hospital wards coincide with their geographical distances. Moreover we set $c_{r,w,\bar{v}} = c_{r,w,\underline{v}} = 0$ for all $r \in R$ and $w \in W$; transportation costs of patients toward home and homecare logical wards are treated as scenario parameters, see below.

Our model considers 10 patients types in set P: seven are obtained from the set of ward specialties found in Regione Lombardia (2020a), to which we have added three additional types representing COVID-19 patients with increasing severity levels of illness (such a distinction is adopted by several governments, see, e.g., National Health Institution, 2020; Son, Lee, & Hwang, 2021):

- patients with mild pneumonia and dyspnea, requiring oxygen through a mask. They can be treated at home in case of lack of human resources and beds in hospitals.
  We label the corresponding type as *COVID-mild*.

- patients requiring treatment in sub-intensive care units (subICU), requiring additional equipment (e.g., Continuous Positive Airway Pressure helmets). They cannot be treated at home.
  We label the corresponding type as *COVID-subICU*;
- patients in intensive care unit (ICU) requiring full ICU equipment and that cannot be treated at home.
  We label the corresponding type as *COVID-ICU*.

Note that our model does not consider asymptomatic and paucisymptomatic patients affected by COVID-19: the former require only isolation surveillance, while the latter can self-treat at home. Each category in P gives rise to a corresponding ward type in S, which also includes two additional types representing the "home" and "homecare" logical wards.

We consider the very initial phase of COVID-19 spread in Spring 2020 in Lombardy region (Italy). That is, in the starting state of the system, the hospitals have no COVID-19 wards and no COVID-19 inpatients. These latter are the only patient types considered as incoming patients. We treat their total quantity and their distribution among hospitals as scenario parameters.

Finally, we consider 10 resources in set R reasonably existing in most Italian hospitals. The minimum quantity of each resource per patient of each type is defined by the Italian legislation (Ministero della Sanità della Repubblica Italiana, 1988), while the initial amount of resources in each hospital ward is computed using data in Regione Lombardia (2020a). The amount of resources available at the global supplier is treated as a scenario parameter.

Scenarios. We vary some parameters to create a set of instances covering multiple realistic scenarios. A summary of such parameters is given in Appendix A, Table 7. Here we describe them in details.

- *Magnitude of the infection* (parameter q_p^{new} for $p \in$ {COVID-mild, COVID-subICU, COVID-ICU}). Let q_p^{new} for $p \in$ {COVID-mild, COVID-subICU, COVID-ICU} be the total number of incoming patients of type p (in the whole system). It will be used to compute parameter $q_{p,h}$ which appears in our model. We consider three values for q_p^{new}:
  - *Baseline scenario:* q_p^{new} is drawn from Regione Lombardia (2020b);
  - *Heavy scenario:* q_p^{new} corresponds to the value in the baseline scenario increased of 25%;
  - *Light scenario:* q_p^{new} corresponds to the value in the baseline scenario decreased of 25%.
- *Distribution of new COVID-19 patients in the hospitals* (parameter $q_{p,h}$ for $p \in$ {COVID-mild, COVID-subICU, COVID-ICU} and $h \in H$). There are two cases:
  - *proportional distribution of new COVID-19 patients:* $q_{p,h} = q_p^{\text{new}} \frac{\sum_{w \in W_h} q^0_{p,w}}{\sum_{w \in W} q^0_{p,w}}$;
  - *real-data distribution of new COVID-19 patients:* we first get the total number sc_ℓ of SARS-CoV-2 positive individuals in each province $\ell \in L$ of Lombardy from (Dipartimento della Protezione Civile, 2020) on the 9th of April 2020 (corresponding to the first peak of 2020 in Lombardy); then we distribute all such individuals as new COVID-19 patients to each hospital h of that province according to the formula $q_{p,h} = q_p^{\text{new}} \frac{\sum_{w \in W_h} q^0_{p,w}}{\sum_{h \in H_\ell} \sum_{w \in W_h} q^0_{p,w}} \frac{\text{sc}_\ell}{\sum_{\ell \in L} \text{sc}_\ell}$, where H_ℓ is the set of hospitals in province $\ell \in L$.
- *resource availability from suppliers* (parameter $q^0_{r,e}$, for $r \in R$ and $e \in E$): we define the amount of resources available to be purchased from supplier as a percentage of the total amount of resources already present in all wards. That is, $q^0_{r,e} = \xi \cdot \sum_{w \in W} q^0_{r,w}$ with $\xi \in \{0, 0.02, 0.05\}$.
- *Maximum transportable distance* (all parameters a_{p,h_1,h_2} and $\hat{d}$). Binary parameters a_{p,h_1,h_2}, used in constraints (2) and (3) to allow the transportation of patients between hospitals are set to value 0 (hence forbidding transport) if the distance between the departure and destination hospital exceeds a threshold value $\bar{d}$. In each considered scenario $\hat{d} = \bar{d}$ and values for both distance types are $\hat{d}, \bar{d} \in$ {50 kilometer, 100 kilometer}.
- *Objective function multiplier for home ward* (parameter $c_{p,w_1,\underline{v}}$ for all $p \in P$ and $w_1 \in W$). $c_{p,w_1,\underline{v}} = \gamma \cdot \max\{c_{p,w_1,w_2} : p \in P, w_1 \in W, w_2 \in W \setminus \{\underline{v}, \bar{v}\}\}$ for $p \in P$ and $w_1 \in W$ with $\gamma \in \{2, 10, 20\}$.

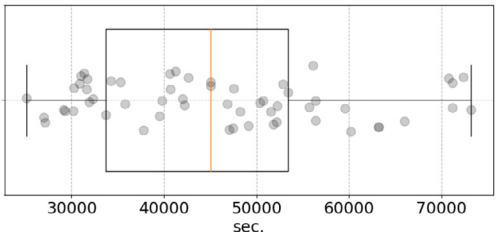


Fig. 3. CPU time MPH.

By combining all cases for the parameter values above we get a total of 108 instances.

Implementation details. The column-generation and the heuristic algorithms of Section 4 have been implemented in C++, using Gurobi 9.5 as LP and MILP solver. We used default values for all parameters of Gurobi. We run our experiments on a machine equipped with an Intel i7 8-core 4.00 gigahertz processor and 32 gigabyte of RAM.

In all our experiments, parameter UNFIXED_WARD_TYPES of the heuristic of Algorithm 1 is set to 2. This low value is justified by recalling that, each time we generate a subset F' of variables to fix in formulation (17), we need to check that F' has not been generated in some previous iteration. Such an operation is time-consuming for higher values of UNFIXED_WARD_TYPES. Moreover, an initial heuristic solution χ^0 was found by fixing wards to their initial type ($s^0(w)$) if their type does not allow hosted inpatients to be discharged, i.e., $\chi_{w,s^0(w)} = 1$ if and only if $a_{s^0(w),\underline{w}} = 0$.

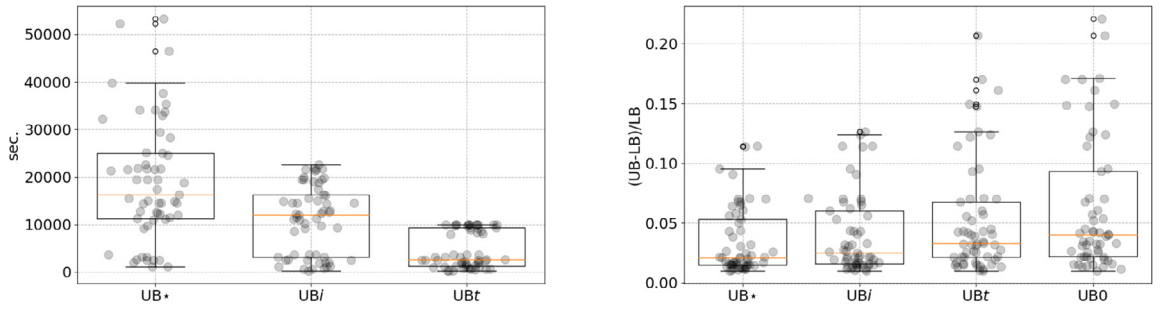
5.1. Computational results

Testing reports infeasibility on all 36 instances with $\xi = 0$; moreover, there are 12 additional infeasible instances corresponding to parameters $\bar{d} = 50$ *kilometer*, $\xi \in \{0.02, 0.05\}$, $\gamma \in \{2, 10, 20\}$ under the real-data distribution of new COVID-19 patients and in both baseline and heavy scenarios. Finally, MPH does not terminate on three additional instances with $\gamma = 20$.[1]

For each of the remaining 57 instances, the best solution obtained by our heuristic is feasible also with respect to constraints (10) in which $\hat{n}_{s,r}$ is initialized from real-world data: our LP-based post-processing is always able to verify the compliance with such constraints and to compute the corresponding values of variables τ. The performance of our algorithms is analysed below on the same set of 57 instances. The exact CPU times, optimality gaps and infeasibility details are reported in Table 8 of Appendix D.

The total CPU times of our algorithm MPH are represented in the boxplot of Fig. 3. In the same figure each gray dot represents the execution of MPH on one of the feasible instances. There is a huge difference between the fastest execution and the slowest one (25,149 seconds vs. 73,153 seconds). The median CPU time value is 45,065 seconds (orange line in the boxplot) which is close to the average CPU time (45,805 seconds). We recall that, as specified in Section 4, Gurobi was not able to load in memory our instances of HRM and was not even able to solve their continuous relaxations.

[1] This is due to memory overflow or to the limit of time and iterations imposed for the resolution of the pricing sub-problems.



(a) Time to reach primal solutions (sec.)

(b) Gap of primal solutions

Fig. 4. Profiling of MPH.

Looking at the computational results more closely we found out that the scenario parameter that mostly influence the CPU time is the distance $\bar{d}$ (and hence the related parameters a_{p,h_1,h_2}). By increasing $\bar{d}$ from 50 kilometer to 100 kilometer the median CPU time increases from 31,730 seconds to 47,543 seconds. Moreover, only 7 out of the 23 instances with $\bar{d} = 50$ kilometer yield a worse CPU time than at least one instance of those with $\bar{d} = 100$ kilometer.[2] For the other parameters we did not observe such strong impact on the CPU times of MPH, hence we omit the corresponding analyses.

Primal solution profiling. We hereby present a profiling of the primal heuristic solutions provided by the iterative runs of VLSNS (line 13 of Algorithm 1), taking into consideration the number of runs with different sets of fixed variables F' and the execution time. Let: UB^0 be the value of the solution χ^0 which initialize MPH; $UB^\star$ be the value of the best solution found by MPH; UB^i be the best primal solution found after i runs of VLSNS; and UB^t be the best primal solution found stopping the execution of MPH after t seconds. In this analysis we set $i = 10$ and $t = 10^4$ seconds (i.e., less than 3 hours). The boxplots in Fig. 4(b) summarize the relative gaps between each upper bound defined above and the lower bound obtained at the end of the CG (value LB), computed as $(UB - LB)/LB$; the boxplots in Fig. 4(a) summarize the CPU times needed to obtain the upper bounds defined above.

First, we analyse the quality of the best solution $UB^\star$ provided by MPH. In Fig. 4(b) we observe that the relative gaps of $UB^\star$ range from 11% to 0.1% with a median value of 2%. Additionally we report that the median relative gap improvement between UB^0 and $UB^\star$ (computed as $(UB^0 - UB^\star)/UB^0$ on each instance) is about 4.7%. These results lead to the conclusion that the combination of our CG algorithm and the VLSNS procedure provides solutions of suitable quality and effectively improves the starting heuristic solution.

Next, we focus on the quality of the solutions produced by MPH by truncating its execution after i runs of VLSNS. The corresponding boxplot in Fig. 4(b) shows that the relative gaps of UB^i are very similar to the ones of $UB^\star$. This suggests that just after 10 runs of VLSNS we get solutions of good quality through our MPH algorithm; indeed, we report that UB^i is strictly better than UB^0 on all 57 instances and that $UB^i = UB^\star$ in 39 instances. Concerning the quality of the solutions obtained by stopping MPH after 10,000 seconds, the median relative gap of UB^t is one percentage point higher than those of $UB^\star$ and UB^i. These solutions improve the starting upper bound UB^0 in 46 out of 57 instances, and in 13 of these we have $UB^t = UB^\star$.

We finally study how much CPU time is actually needed to obtain the primal solutions corresponding to $UB^\star$, UB^i and UB^t. The first boxplot in Fig. 4(a) shows that the median time needed to find $UB^\star$ is around 16,000 seconds, with a maximum of ∼53,000 seconds; the average time to get $UB^\star$ is ∼19,000 seconds. This is much lower than the ∼45,000 seconds required to terminate the MPH algorithm in median. The remaining two boxplots of Fig. 4(a) reports the quartiles of the CPU times for obtaining UB^i and UB^t respectively. In median, we need ∼12,000 seconds to find UB^i and ∼4,500 seconds to find UB^t.

Discussion. Our primal solution profiling yields that the MPH algorithm is suitable to support tactical planning decisions demanded by realistic HRM instances: a truncated version is able to find good solutions in less than a quarter of the time needed to complete the column generation procedure. Therefore, although this latter takes high CPU times (about 12 hours in average), its use is still adequate to tackle epidemic emergencies that last from weeks to months, such as that of COVID-19.

Our experiments also highlight that only the maximum transport distance of the patients has a relevant effect on the CPU time of our algorithms. We impute this behavior to the fact that small values of $\bar{d}$ set to 0 more parameters a_{p,h_1,h_2}, which in turn, through constraints (2) and (3), fixes to 0 the corresponding variables in our model: $\bar{d} = 50$ kilometer sets to 0 the 58% of μ and π variables, while $\bar{d} = 100$ kilometer sets to 0 the 28% of them. Then, $\bar{d} = 50$ kilometer corresponds to a smaller solution space of the CG, which terminates in less time. [3]

5.2. Comparison with existing methods

We stress that our methods are designed to handle the HRM in all its features: as soon as some of them are dropped, simpler and arguably more efficient methods could be employed. For instance, if location decisions are fixed, our model becomes similar to that of Sun et al. (2014); if all resources belong to the same commodity, and facilities are homogeneous, the effective methods of Corberán et al. (2020) can be used. In fact, we are not aware of attempts in the literature considering these features simultaneously. A natural question is instead whether existing models for rich FLA problems can be exploited to provide (even approximate) solutions to our HRM.

Indeed, the model for the FLA problem with capacity transfer (FLA-CT in the remainder) of Corberán et al. (2020) is suitable to be adapted to our HRM by penalizing the assignment of multiple types to the same facility and treating resources in a surrogate way, minimizing infeasibilities arising due to incompatible resource usage in post-processing. We have experimented on different adaptation variants, finding one of them especially suitable for our purpose; full details are reported in Appendix E.

[2] A qualitative view of the change in the CPU time with respect to this parameter is given in Appendix D, Fig. 5.

[3] Additional qualitative view of this effect is provided in Appendix D, Fig. 6.

Table 2
Results of FLA-CT vs. MPH.

		MPH		FLA-CT Corberán et al. (2020)			
$H\%$	$\|W\|$	time UB* (seconds)	time CG (seconds)	abs. viol. (1)	forbidden resource usage (%)	time (seconds)	unsolved instances
10	62	6.03	112.75	4	20%	56.88	0
25	147	52.94	1276.88	11.63	44%	152.50	0
50	272	400.04	4853.00	10.63	47%	1758.63	0
75	392	1057.22	12553.00	10.38	50%	4614.75	0
100	483	13599.88	38053.25	7.5	48%	6620.33	2

We experiment on a subset of our instances. We set a baseline scenario with parameters $\gamma = 2$, $\xi = 0.05$, $\bar{d} = 50$ and 'proportional'-'light' distribution of new COVID-19 patients. We experiment on this baseline scenario and all configurations that change the value of exactly one parameter starting from the baseline, for a total of 8 instances. Moreover, for each instance, we create sub-instances selecting a subset of hospitals with their corresponding wards. That is, on top of the complete instances, we consider sub-instances with 10%, 25%, 50% and 75% of hospitals. In Table 2 we summarize our comparison. Full computational results are reported in Appendix E, Table 11. Columns in first block contain the size of the instance, in terms of percentage of hospitals ('H%') and number of wards ('$|W|$'). Second block contains the computing time for the MPH algorithm to find the best primal solution ('time UB*') and to complete the CG ('time CG'). The third block refers to the FLA-CT algorithm of Corberán et al. (2020), with the number of times an additional type is assigned to a single ward (the absolute violation of constraint (1), 'abs. viol. (1)'), the percentage of resources that are used in the solution of FLA-CT but which violate constraints (5) or (6) ('forbidden resource usage (%)'), the computing time ('time') and the number of unsolved instances. Each row reports average value over 8 instances. The time needed by MPH to find the best solution ('time $UB^\star$') is less than the time needed by FLA-CT to finish execution (with the exception of the case of the complete instance '$H\%$'$= 100$), while the execution time of FLA-CT is less than the time to complete CG execution of MPH. However, when considering the complete instance (case with '$H\%$'$= 100$), in 2 cases out of 8 FLA-CT was not able to provide a solution for 'out-of-memory' errors; that is, instances of real-world size such as those used in our experiments currently require decomposition techniques to be handled, as in MPH algorithm.

Details on the computation of infeasibilities are provided in Appendix E. The amount of violations is not negligible, with up to 50% of resources that are used in the solution of FLA-CT but whose use is forbidden for HRM. As a consequence, additional processing is required to use FLA-CT solutions for the original HRM problem, while MPH guarantees feasibility of its solutions.

5.3. Multi-objective optimization

We experiment the multi-objective optimization scheme presented in Algorithm 2 on the subset of instances defined in the preceding subsection. In Table 3 we present a summary of computational results, averaged over the 8 instances. The complete set of results is presented in Appendix D, Table 9.

First we check the distance between utopia UB^U and the value $UB^\star$ provided by our algorithm MPH, and the distance between the utopia and the lower bound computed at the end of CG for its computation (LB^U). The gap between the utopia and its lower bound ($(UB^U - LB^U)/LB^U$) is fairly low (2% for C^E and 3% for C^{QoS}). In terms of absolute values, C^E is much smaller than C^{QoS} (around 0.1%); that yields high relative gap between its utopia and the nadir ($(UB^\star - UB^U)/UB^U$). Instead, the gap between the utopia and the nadir for C^{QoS} is very low (around 1%).

Table 3
Average value of computational results of multi-objective optimization experiments.

	C^E	C^{QoS}
utopia UB^U	39.50	37976.00
nadir $UB^\star$	83.15	38343.90
$(UB^\star - UB^U)/(UB^U)$	1.936	0.012
$(UB^U - LB^U)/(UB^U)$	0.024	0.032
time $UB^\star$ (seconds)	38053.25	38053.25
time UB^U (seconds)	6094.75	43591.00
time 1st ϵ-constr. iteration (seconds)	18579.5	25927.38
CG iter. 1st ϵ-constr. iteration	185.5	257.88
time ϵ-constr. reoptimization (seconds)	1601.81	2475.56
CG iter ϵ-constr. reoptimization	2.8	4.7

It is therefore not surprising that no new solution is found in the 8 experiments using C^E as objective function, as any such solution should fall into that narrow gap. It was instead possible to find non dominated solutions on 2 of the 8 experiments using C^{QoS} as objective. Concerning the computational profiling: the first visited grid point requires an execution time and a number of CG iterations similar to those required by the setting without threshold values; on the other hand, the subsequent grid point requires only a small amount of time and CG iterations, thanks to the warm start described in Section 4.3. Overall, in terms of problem structure, using C^E as objective and limiting C^{QoS} by a constraint, results in faster runs.

6. Conclusion

In this paper we have studied the HRM, a tactical facility location-allocation problem applied to the reorganization of medical resources and hospitals in large geographic areas. In the HRM the combinatorial structure of standard facility location-allocation problems is enriched with constraints and features emerging from the management of medical resources in real world applications, such as times and costs of transportation of resources and patients, incompatibility between different types of patients in a same ward, need of specific wards and staff for each patient type, etc.

To obtain primal solutions to the HRM we have developed a mathematical programming heuristic, intertwining column-generation with very large-scale neighborhood search. Our heuristic is based on a MILP model for the HRM whose objective function balances between the quality of service delivered to the patients and the economic costs resulting from the health system reorganization. The flexibility of our heuristic allowed us to embed it in a multi-objective optimization algorithm inspired by the augmented ϵ-constraint approach, where the quality of service and the economic costs are minimized separately. Both the mathematical programming heuristic and the multi-objective approach provide valid lower bounds through the column generation mechanism, allowing an evaluation of the primal solutions quality.

To show the practical applicability of our algorithms we tested them on instances inspired by the COVID-19 emergency that occurred in Lombardy (Italy) at the beginning of 2020. Data collec-

tion itself was not trivial, as real data is available only by aggregating multiple heterogeneous public repositories of Italian institutions, which are partially unstructured. We openly release them as structured instances, to foster reproducibility and further research.

We have performed scenario-based experiments, by considering several estimations of the COVID-19 incoming patients at the epidemic peak and other parameters related to the resource availability, to geographical constraints and to the quality of service.

The experimental results indicate that our mathematical programming heuristic is successful in terms of both computational times and solutions quality: it detects infeasible instances quickly and solves most of feasible instances employing a CPU time which is adequate for tactical purposes; the obtained solutions are of good quality, as we have shown by estimating their optimality gaps, based on the lower-bound provided by the column generation algorithm.

Concerning the task of optimizing the two objectives independently, in a multi-objective approach we found that (a) our aggregation method is effective in producing solutions dominating a large portion of the potential space of Pareto-optimal solutions (b) our column generation method fits well in this context, given its strong reoptimization potential when ϵ-constraints iteratively change.

We examined several research directions for future works. First, we would study exact algorithms for solving the HRM. A branch-and-price algorithm is the most natural extension in this direction, since it is based on column-generation techniques. Second, we would integrate our tactical decision-support model with tools for estimating epidemic peaks (epidemiological models, data analysis, etc.). Finally, we would test the approach resulting from the previous two steps also in a rolling-horizon setting, in which the reorganization of the health system is performed at successive moments of a same epidemic spread.

Acknowledgments

The authors wish to thank Pier Giorgio Villani and Giovanni Righini for their invaluable insights into the specific health system domain, and their suggestions on its modeling features. Three anonymous reviewers and the editor shared their valuable comments, helping us to substantially improve the paper. The work has been partially supported by Regione Lombardia, under the Regional Operative Program “European Regional Development Fund 2014–2020”, grant ID R1.2020.0002349, project COD-19.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.ejor.2022.11.024

References

Ahmadi-Javid, A., Seyedi, P., & Syam, S. S. (2017). A survey of healthcare facility location. *Computers and Operations Research, 79*, 223–263.
Altay, N., & Green, W. G., III (2006). OR/MS research in disaster operations management. *European Journal of Operational Research, 175*(1), 475–493.
Andersen, A. R., Nielsen, B. F., & Reinhardt, L. B. (2017). Optimization of hospital ward resources with patient relocation using Markov chain modeling. *European Journal of Operational Research, 260*(3), 1152–1163.
Boonmee, C., Arimura, M., & Asada, T. (2017). Facility location optimization model for emergency humanitarian logistics. *International Journal of Disaster Risk Reduction, 24*, 485–498.
Carr, S., & Roberts, S. (2010). Planning for infectious disease outbreaks: Ageographic disease spread, clinic location, and resource allocation simulation. In *Proceedings of the 2010 winter simulation conference* (pp. 2171–2184). IEEE.
Corberán, A., Landete, M., Peiró, J., & Saldanha-da Gama, F. (2020). The facility location problem with capacity transfers. *Transportation Research Part E: Logistics and Transportation Review, 138*, 101943.
Cunningham, A. A., Daszak, P., & Wood, J. L. N. (2017). One health, emerging infectious diseases and wildlife: Two decades of progress? *Philosophical Transactions of the Royal Society B: Biological Sciences, 372*(1725), 20160167.
Dasaklis, T. K., Pappis, C. P., & Rachaniotis, N. P. (2012). Epidemics control and logistics operations: A review. *International Journal of Production Economics, 139*(2), 393–410.
Demaine, E. D., Hajiaghayi, M., Mahini, H., Sayedi-Roshkhar, A. S., Oveisgharan, S., & Zadimoghaddam, M. (2009). Minimizing movement. *ACM Transactions on Algorithms (TALG), 5*(3), 1–30.
Desrosiers, J., & Lübbecke, M. E. (2005). A primer in column generation. In *Column generation* (pp. 1–32). Springer.
Dipartimento della Protezione Civile, P. (2020). COVID-19 Italia - Monitoraggio situazione. Accessed: 2021-01-30 https://github.com/pcm-dpc/COVID-19.
Ekici, A., Keskinocak, P., & Swann, J. L. (2008). Pandemic influenza response. In *2008 winter simulation conference* (pp. 1592–1600). IEEE.
Gagliano, A., Villani, P. G., Manelli, A., Paglia, S., Bisagni, P. A. G., Perotti, G. M., et al., (2020). COVID-19 epidemic in the middle province of Northern Italy: Impact, logistics, and strategy in the first line hospital. *Disaster Medicine and Public Health Preparedness, 14*(3), 372–376.
Gurobi Optimization, L. (2021). Gurobi optimizer reference manual. http://www.gurobi.com.
Her, M. (2020). Repurposing and reshaping of hospitals during the COVID-19 outbreak in South Korea. *One health, 10*, 100137.
Huang, R., Kim, S., & Menezes, M. B. C. (2010). Facility location for large-scale emergencies. *Annals of Operations Research, 181*(1), 271–286.
Jain, V., Duse, A., & Bausch, D. G. (2018). Planning for large epidemics and pandemics: Challenges from a policy perspective. *Current Opinion in Infectious Diseases, 31*(4), 316–324.
Jia, H., Ordóñez, F., & Dessouky, M. (2007a). A modeling framework for facility location of medical services for large-scale emergencies. *IIE Transactions, 39*(1), 41–55.
Jia, H., Ordonez, F., & Dessouky, M. M. (2007b). Solution approaches for facility location of medical supplies for large-scale emergencies. *Computers and Industrial Engineering, 52*(2), 257–276.
Klose, A., & Drexl, A. (2005). Facility location models for distribution system design. *European Journal of Operational Research, 162*(1), 4–29.
Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization, 26*(6), 369–395.
Mavrotas, G. (2009). Effective implementation of the ε-constraint method in multi-objective mathematical programming problems. *Applied Mathematics and Computation, 213*(2), 455–465.
McMichael, A. J. (2013). Globalization, climate change, and human health. *New England Journal of Medicine, 368*(14), 1335–1343.
Melo, M. T., Nickel, S., & Da Gama, F. S. (2006). Dynamic multi-commodity capacitated facility location: A mathematical modeling framework for strategic supply chain planning. *Computers and Operations Research, 33*(1), 181–208.
Melo, M. T., Nickel, S., & Saldanha-Da-Gama, F. (2009). Facility location and supply chain management—A review. *European Journal of Operational Research, 196*(2), 401–412.
Meschi, T., Rossi, S., Volpi, A., Ferrari, C., Sverzellati, N., Brianti, E., … Ticinesi, A. (2020). Reorganization of a large academic hospital to face COVID-19 outbreak: The model of Parma, Emilia-Romagna region, Italy. *European Journal of Clinical Investigation, 50*(6), e13250.
Ministero della Sanità della Repubblica Italiana (1988). Decreto 13/09/1988: Determinazione degli standards del personale ospedaliero. https://www.gazzettaufficiale.it/eli/id/1988/09/24/088A3830/sg. Accessed: 2021-01-30.
National Health Institution (2020). Clinical Spectrum of SARS-CoV-2 Infection. Last checked on 2021-05-21 https://www.covid19treatmentguidelines.nih.gov/overview/clinical-spectrum/.
Nicola, M., Alsafi, Z., Sohrabi, C., Kerwan, A., Al-Jabir, A., Iosifidis, C., et al., (2020). The socio-economic implications of the coronavirus and COVID-19 pandemic: A review. *International Journal of Surgery, 78*, 185–193.
Parra, M. A., Terol, A. B., Gladish, B. P., & Uría, M. V. R. (2005). Solving a multiobjective possibilistic problem through compromise programming. *European Journal of Operational Research, 164*(3), 748–759.
Pishnamazzadeh, M., Sepehri, M. M., Panahi, A., & Moodi, P. (2021). Reallocation of unoccupied beds among requesting wards. *Journal of Ambient Intelligence and Humanized Computing, 12*(1), 1449–1469.
Premoli, M., Barbato, M., & Ceselli, A. (2021). COVID-19 Hospital Resource Management Dataset - Lombardy 2020 - replication data. 10.13130/RD_UNIMI/WWUZIJ
Raghavan, S., Sahin, M., & Salman, F. S. (2019). The capacitated mobile facility location problem. *European Journal of Operational Research, 277*(2), 507–520.
Regione Lombardia (2020a). Letti per struttura sanitaria di ricovero. https://www.dati.lombardia.it/Sanit-/Letti-per-struttura-sanitaria-di-ricovero/m2eh-mypv/data. Accessed: 2021-01-30.
Regione Lombardia (2020b). Regione Lombardia Dashboard COVID-19. Last checked on 2021-05-21 https://www.regione.lombardia.it/wps/portal/istituzionale/HP/servizi-e-informazioni/cittadini/salute-e-prevenzione/coronavirus/dashboard-covid19/.
Regione Lombardia (2020c). Strutture di ricovero e cura. https://www.dati.lombardia.it/Sanit-/Strutture-di-ricovero-e-cura/teny-wyv8/data. Accessed: 2021-01-30.
Scarfone, R. J., Coffin, S., Fieldston, E. S., Falkowski, G., Cooney, M. G., & Grenfell, S. (2011). Hospital-based pandemic influenza preparedness and response: Strategies to increase surge capacity. *Pediatric Emergency Care, 27*(6), 565–572.
Shuman, E. K. (2010). Global climate change and infectious diseases. *New England Journal of Medicine, 362*(12), 1061–1063.

Silal, S. P., et al., (2021). Operational research: A multidisciplinary approach for the management of infectious disease in a global context. *European Journal of Operational Research, 291*(3), 929–934.
Son, K.-B., Lee, T.-j., & Hwang, S.-s. (2021). Disease severity classification and COVID-19 outcomes, Republic of Korea. *Bulletin of the World Health Organization, 99*(1), 62.
Stilianakis, N., & Consoli, S. (2013). Operations research in disaster preparedness and response: The public health perspective. *Technical report JRC, Report EUR 25763 EN*. Publications Office of the European Union, Luxembourg.
Sun, L., DePuy, G. W., & Evans, G. W. (2014). Multi-objective optimization models for patient allocation during a pandemic influenza outbreak. *Computers and Operations Research, 51*, 350–359.
Thomson, S., Nunez, M., Garfinkel, R., & Dean, M. D. (2009). Efficient short term allocation and reallocation of patients to floor of a hospital during demand surges. *Operations Research, 57*(2).
Villani MD, P. G. (2020). ASST Lodi, Department of Emergency and Critical Care Unit of Anesthesia and Resuscitation, Lodi, Italy. Private communication.
Zuccotti, G. V., Bertoli, S., Foppiani, A., Verduci, E., & Battezzati, A. (2020). COD19 and COD20: An Italian experience of active home surveillance in COVID-19 patients. *International Journal of Environmental Research and Public Health, 17*(18), 6699.