OR Spectrum manuscript No.
(will be inserted by the editor)

Effective Patient Prioritization in Mass Casualty Incidents using Hyperheuristics and the Pilot Method

Carlos Cotta

Received: date / Accepted: date

Abstract Whenever a mass casualty disaster takes place, the medical infrastructure available has to deal with a surge in the number or patients severely ill or injured. Using triage methods casualties have to be prioritized to receive health care in a limited-resource scenario. Aiming to do the greatest good to the greatest number of people, it has to be determined how to make the best use of these resources. This constitutes a very complex task that has to consider issues such as the current number of casualties, their lifetime expectancy, their resource consumption, etc. We approach this task within the framework of the pilot method and hyperheuristics. We show how these metaheuristics can effectively manage a number of simpler heuristics, providing improved results on an ample set of simulated problem scenarios. An exhaustive empirical evaluation analyzes the influence on performance of factors such as the total number of casualties, the severity of their medical condition, the treatment time, the number of resources available, or the number of triage classes.

Keywords Mass casualty incident · Triage · Hyperheuristics · Pilot method

1 Introduction

A disaster is a catastrophic event that seriously disrupts the normal functioning of society at a scale which may vary depending of its magnitude [1]. In the aftermath of a disaster, society has to cope with the damage infringed, both from the material and the humanitarian perspective. The latter is actually one

This work is supported by Spanish Ministerio de Ciencia e Innovación under project TIN2008-05941 (Project NEMESIS).

C. Cotta
ETSI Informática, Universidad de Málaga, Campus de Teatinos, 29071 Málaga, Spain
Tel.: +34-952-137158
Fax: +34-952-131397
E-mail: ccottap@lcc.uma.es

of the most tragic aspects of a disaster, and dealing with it can undoubtedly constitute a major challenge: a quick response is needed to deliver humanitarian relief and appropriate medical care to the victims of the disaster, in a scenario in which basic infrastructures for communication and transportation may be greatly affected. Needless to say, this involves being able to manage adequately all available resources [2], in particular when these are scarce with respect to the number of casualties and the severeness of their injuries.

Among the numerous problems arising in a situation as described – involving transportation logistics [3], medical routing [4], facility location [5] etc. – we will focus on the decision-making underlying the distribution of medical care to casualties. In this sense, mass casualty disasters require a paradigm change from the standard approaches to emergency room care in which available medical resources are not overwhelmed by the sporadic arrival of casualties [6]. Quite on the contrary, in a mass casualty scenario health care demand typically exceeds hospital resources (e.g., imaging devices, life-support systems, operating rooms, etc.). It is thus crucial to make the most effective use of these limited resources.

The term *triage* is used to denote the mentioned decision-making process for distributing medical resources among patients [7–9]. Roughly speaking, triage involves sorting patients in different categories according to their medical condition, and prioritizing treatment among them. A more detailed overview on triage methods will be provided in Section 2. A recurrent theme in triage systems is ensuring the maximal benefit from the limited medical resources [10]. We approach this problem from an utilitarian perspective in which the goal is attaining the greatest good for the greatest number of people [11, 12]. More precisely, we consider the problem of prioritizing patients in order to maximize the expected number of survivors. This is done on the basis of available information on survival probabilities and how these change over time for each patient category. This problem will be formalized in Section 3.1.

We approach this patient prioritization problem via metaheuristics. To the best or our knowledge, this constitutes a novel application domain for these techniques; to be precise we consider hyperheuristics [13–16] and the pilot method [17–19]. These techniques will exploit heuristics recently defined for scheduling impatient jobs [20], and will be described in Section 3.3. We have conducted an extensive experimental evaluation of these techniques on a large number of simulation scenarios intended to capture mass casualties incidents of different severity. Given the nature of the problem, any gain that can be attained –even if small– is very valuable. As it will be shown in Section 4, this is generally the case for metaheuristics, which compare favorably in general to existing heuristic policies. The results also provide some insights on the sensitivity of these heuristics to different features of the disaster scenarios (e.g., number of resources available, severity of patient conditions, overall number of patients, etc.). This information yields useful hints on the strengths and limitations of each technique.

2 Background and Related Work

As mentioned in Section 1, when a mass casualty incident (MCI) takes place there is a sudden and serious disproportion between the resources required by the casualties and the resources that are available. Focusing on critically ill or injured patients, a surge in their number in the aftermath of a MCI will overwhelm the capacity of hospitals and critical care units, decreasing their response capability [21]. In this context, the notion of surge capacity is precisely defined as the ability to cope with a sudden, unexpected increase in patient volume beyond the present capacity of the facility [22].

Even though most medical facilities have a certain surge capacity, peak demand of limited resources (X-ray devices, mechanical ventilators, operating rooms, etc.) will lead to dramatic situations in which these resources must be rationed and directed to patients who will benefit most from them [12]. In this scenario, the needs of the community as a whole stand above those of individuals considered in isolation. The implications of this change of paradigm are manifold, and include the temporary adjustment of the standard of care for all patients [23], directing resources to patients to whom these will be most effective. The process of sorting and prioritizing patients is termed triage.

Leaving aside the profound ethical issues surrounding triage in the aftermath of a MCI [24–26], its actual technical implementation is complex. Currently, there are about a dozen mass-casualty triage systems in use around the world [27]. These triage methods sort patients into groups according to a certain number of medical indicators (pulse rate, breath status, etc.). For example, one of the most commonly used systems is the *Simple Triage And Rapid Treatment* (START) system [28]. This system is aimed at providing rescuers with the ability of classifying patients in less than 60 seconds into four classes: green (delayed care), yellow (urgent care), red (immediate care), black ('expectant' or dead). Other triage systems may differ in the set of medical variables considered, or in the resulting patient groups (e.g., adding a blue/violet class for likely expectant patients). For a comparative of triage methods, the reader is referred to [29].

Triage does not end with the classification of patients into groups as sketched above. Actually, that is just the first step of the process, which can be described as field triage or primary triage. Further re-examination and prioritization can take place at different points of the medical care chain, such as at hospital arrival or at the intensive care unit level. With the goal of distributive justice in mind, patient prioritization may not necessarily equate to the severeness of their medical condition though. Certainly, less severely injured patients can better tolerate delays and/or some degree of suboptimal care [30] (the principle of 'minimal acceptable care' [31]). Likewise, START expectant category is meant to leave out of further consideration those patients who will not survive even with maximal resuscitative effort [27] (of course, such patients are entitled to receive palliative treatment and comforting measures to preserve their dignity). However, in some cases it has been suggested that priority should be given to moderate severity patients rather than to those of the greatest severity [6]. Such a decision is motivated by the different profile of resource consumption by patients in different groups – check, e.g., [32].

Health care officials thus face complex decisions that need to be addressed not just considering the criticality of patient conditions, but also the number of patients –and their corresponding health status– in need of using the same medical resources [23, 33]. This issue has been recently addressed by Argon et al. [20], by analyzing the conditions in which a state-independent policy (i.e., a policy that does not take into account the number of patients in each triage class) can be optimal. They consider a single-server system (that is, one single resource used in mutual exclusion and non-preemptively by waiting patients, e.g., an operating room). It is shown that if patients can be ordered such that those in a most urgent life-threatening situation also require less time to be serviced, then the optimal policy will give priority to these. However, in a most typical scenario in which patients of greatest severity also require longer use of the server, the optimal policy has a complex structure that depends on the system state. In a scenario in which both lifetimes and operation times are exponentially distributed the system is memoryless. Hence the optimal policy is time-independent and only needs to consider the number of patients in each triage class. In a more general (and realistic) situation in which lifetimes follow a different distribution (e.g., Weibull [34]), time is however a defining characteristic of the system state as well, thus greatly increasing the complexity of the problem. This can be further aggravated if there are more than one shared resource –e.g., multiple operating rooms– as we will consider here. In this context, we pose the use of two metaheuristic approaches –hyperheuristics and the pilot method– to approach this prioritization problem. These metaheuristics will use as internal lower level heuristics both state-independent and state-dependent heuristics defined in the literature [20, 35].

3 Solving the Patient Prioritization Problem

In order to tackle the problem outlined before, let us firstly formulate it in a more precise way. Subsequently, we will describe some heuristics for the problem that will pave the way to define our metaheuristic approaches.

3.1 Problem Formulation

As mentioned in Section 2, field triage methods classify casualties into one of several groups on the basis of a quick assessment of several health variables. We assume this classification clusters patients into tiers $c_1, \cdots, c_k$, such that c_i patients have a more critical condition than those of c_j, $j > i$. Such criticality is modelled by means of a lifetime expectancy, which we assume to be Weibull-distributed. The Weibull distribution is commonly used in survival analysis to model the lifetime of individuals or the time-to-failure in mechanical devices [36]. One of the most salient features of the Weibull distribution is the fact that it allows generalizing the exponential distribution: while the latter corresponds to a constant hazard rate (hence the memoryless property), the Weibull distribution can model an increasing, constant, or decreasing hazard rate depending on a certain shape parameter α_i. We consider $\alpha_i > 1$ and therefore the hazard rate increases with time, in accordance to the aggravated state of casualties pending medical treatment. In this case, the mean lifetime is given by $\beta_i \Gamma(1 + 1/\alpha_i)$, where $\Gamma(\cdot)$ is the gamma function and β_i is the scale parameter. As to service times, which we will refer to as operation times henceforth, we consider two scenarios. We will initially consider patients in each class c_i require a deterministic time τ_i to be treated. While simplified, this assumption is however consistent with a 'damage control' situation, in which rapid and abbreviated care is given in the operating room until the MCI overload recedes [6]. In such a situation, operation times may not greatly fluctuate. In any case, we also analyze a second scenario in which operation times are stochastic. To be precise, our model considers that any operation needs a minimum time τ_i, and can have an excess time which for simplicity is assumed to be exponentially-distributed with parameter $1/(\eta_i \tau_i)$, where η_i is an additional class parameter.

Given the above parameters the objective is to take decisions online, to determine from which class the next patient to be operated will be taken, so as to finally maximize the number of patients treated – or equivalently, to minimize the number of patients dead while waiting for treatment. We assume that there are ϱ identical operating rooms available, and therefore decisions are taken any time one of these operating rooms becomes available.

3.2 Basic Heuristics

Heuristics for patient prioritization can be classified as state-dependent and state-independent. The latter are arguably simpler, since they only consider the lifetime estimates and operation times, but not the number of patients in each class. Among these, we have considered the following:

- Time Critical First (TCF): each time t a decision has to be taken, classes are sorted according to decreasing values of their updated abandonment rates $r_i(t)$ (the abandonment rate being the reciprocal of the mean remaining lifetime). Subsequently, a patient of the first non-empty class is taken. Following [20], updated abandonment rates are computed as follows:

$$r_i(t) = \frac{\alpha_i e^{-t'}}{\beta_i \Gamma(1/\alpha_i, t')} \tag{1}$$

  where $t' = (t/\beta_i)^{\alpha_i}$, and $\Gamma(a, b) = \int_b^\infty u^{a-1} e^{-u} du$ is the incomplete gamma function.

- The $r\mu$ heuristic: this heuristic is due to [35], and sorts classes by decreasing values of $r_i \mu_i$, where r_i is the updated abandonment rate computed at time t as before, and μ_i is the service rate (the reciprocal of τ_i).

In addition to these heuristics, two state-dependent policies defined in [20] are considered:

- Triangular heuristic (T): considering a two-class problem, the T heuristic gives priority to class c_1 if

$$\frac{(x_1-1)r_1+x_2r_2}{\mu_1} \leqslant \frac{x_1r_1+(x_2-1)r_2}{\mu_2} \tag{2}$$

  where r_i, μ_i are defined as above, and x_i is the number of patients in c_i. This heuristic is termed triangular because Eq. (2) –along with $x_1, x_2 > 0$ (otherwise no heuristic decision is required)– defines a right triangle in (x_1, x_2)-space. Note that this heuristic can be regarded as a greedy selection procedure picking the class that minimizes the mean number of impatient deaths during operation. We have therefore generalized it to scenarios with more than two classes as selecting the class i that minimizes d_i given by:

$$d_i = \frac{1}{\mu_i}\left(-r_i + \sum_j x_j r_j\right) . \tag{3}$$

  This quantity can be actually seen as the mean number of impatient deaths in all classes when a patient from class i is taken to the operation room.

- Rectangular heuristic (R): related to the previous heuristic, this policy assumes $r_1 > r_2$ and $\mu_1 < \mu_2$, and defines two threshold values:

$$T_1 = \frac{\mu_2(r_1-r_2)}{r_1(\mu_2-\mu_1)} \quad \text{and} \quad T_2 = \frac{\mu_1(r_1-r_2)}{r_2(\mu_2-\mu_1)} . \tag{4}$$

  T_1 (resp. T_2) is obtained by plugging $x_1 = T_1$ and $x_2 = 1$ (resp. $x_1 = 1$, $x_2 = T_2$) in Eq. (2) and solving it as an equality, hence obtaining the coordinates of the endpoints of the triangle hypotenuse. Class c_1 patients are selected if, and only if, $1 \leqslant x_1 \leqslant T_1$ and $1 \leqslant x_2 \leqslant T_2$ (thus defining a rectangle in (x_1, x_2)-space by doubling the triangle defined by the T heuristic; this simple structure is an advantage of this heuristic). Note that unlike the T heuristic, the R heuristic is not directly generalizable to more than two patient classes.

These heuristics have been used as low-level heuristics (LLH) in the metaheuristic approaches defined next.

3.3 Metaheuristics Approaches

The basic heuristics defined in the previous section provide fast yet in general myopic decision procedures. In order to alleviate the 'locality' of the decision-making procedure and obtain globally better solutions we need to add a metaheuristic layer to provide higher-level guidance and escape from greedy traps.

We have approached this using both hyperheuristics [13–16] and the pilot method [17–19].

Starting with the latter, the pilot method can be defined as a tempered greedy method [18, 19] that looks ahead by using a LLH as *pilot*, that is, to obtain an objective-value test of the goodness of each possible choice. To describe the deployment of this method on the prioritization problem, let us consider any prioritization policy Ξ (such as any of those described in the previous subsection) be defined as a function $\textsc{Choice}{-}\Xi(\mathbf{x}, \mathbf{P}, \mathbf{t})$. This function takes as parameters the whole system state: the number of patients $\mathbf{x}$ in each class, the distribution parameters $\mathbf{P}$ defining each class, and the times $\mathbf{t}$ at which each of the operating rooms will be available. Regarding the latter, these times are known in advance in the first scenario in which operation times are deterministic; in the second scenario in which these times are stochastic, an approximation can be used (we consider the minimum operating time τ_i as an optimistic estimation in this case). This function returns the class index from which a patient must be picked, given the system state that is passed as input.

Now, let $\textsc{Update}(\mathbf{x}, \mathbf{P}, \mathbf{t}, j)$ be a procedure that assumes that at time t_γ, where $\gamma = \arg\min\{t_i \mid 1 \leqslant i \leqslant \varrho\}$ a patient from class j is taken to the operating room, and updates the system state accordingly. This involves updating the time the corresponding operating room will be available again ($t_\gamma \leftarrow t_\gamma + \tau_j$), decreasing by one the number of patients in class j, and recomputing the expected number of survivors in each patient class by the time t' the next operating room becomes available. For Weibull-distributed survival times, the probability of a patient surviving up to time t_1 given that he survived up to time t_0 is

$$p(t_0, t_1, \alpha, \beta) = e^{-[(t_1/\beta)^\alpha - (t_0/\beta)^\alpha]} \tag{5}$$

where α and β are respectively the shape and scale parameters as mentioned in the previous section. The expected number of survivors is then computed by multiplying the actual number of patients in each class by their survival probability (calculated using $t_0 = t_\gamma$, $t_1 = t'$, and the corresponding distribution parameters), rounding to the nearest integer.

Finally, let $\textsc{Construct}{-}\Xi(\mathbf{x}, \mathbf{P}, \mathbf{t})$ be a function that takes as input the system state, simulates it to completion using $\textsc{Choice}{-}\Xi$ as decision-making procedure, and returns the total number of patients operated:

```
function CONSTRUCT−Ξ(x, P, t) : ℕ;
begin
    ω ← 0;
    while Σ_i x_i > 0 do
        j ← CHOICE−Ξ(x, P, t);
        ω ← ω + 1;
        UPDATE(x, P, t, j);
    end while
    return ω;
end
```

Now, given a certain heuristic Ξ, let us define policy PILOT(Ξ) as given by the following choice function:

```
function Choice−Pilot(Ξ)(x, P, t) : ℕ;
begin
    σ ← {i | x_i > 0};
    for i ∈ σ do
        x′ ← x; t′ ← t;
        Update(x′, P, t′, i);
        ζ_i ← Construct−Ξ(x′, P, t′);
    end for
    return arg max{ζ_i | i ∈ σ};
end
```

As it can be seen, CHOICE−PILOT(Ξ)($\cdot$) is a higher-order function that uses CONSTRUCT−Ξ to obtain an indication of the goodness of making each of the possible choices at a given instant (i.e, a projection of the number of patients treated; in case of ties, the most critical class is taken). This way, choices are more informed since they rest on actual objective values rather than on myopic measures. The quality of the pilot Ξ is crucial in the performance of the algorithm though. This will be empirically analyzed in the next section.

The second metaheuristic approach considered is based on hyperheuristics. These can be defined as higher level heuristics that manage a set of LLHs (of cardinality greater than one), using only limited problem information [15]. Basically, the hyperheuristic decides at each instant which of the available LLHs will be used. The underlying idea is thus making combined use of several LLHs, so that by making appropriate choices it is possible to exploit their strengths and compensate their weaknesses [14]. Such choices can be done in a variety of ways: at random, using some greedy measure, using some kind of machine learning mechanism, or even using a full-fledged metaheuristic to optimize the sequence of LLHs invocations. In this case, the mechanism that best suits the needs of fast online decision-making is a greedy selection method. More precisely, let $\mathbf{\Xi} = \{\Xi_1, \cdots, \Xi_h\}$ be a set of LLHs defined in Section 3.2. Then, let us define CHOICE−HYPER($\mathbf{\Xi}$)($\cdot$) as follows:

```
function Choice−Hyper(𝚵)(x, P, t) : ℕ;
begin
    σ ← {Choice−Ξ_i(x, P, t) | Ξ_i ∈ 𝚵};
    if |σ| = 1 then
        return [σ]; // returns the only element in σ.
    else
        for i ∈ {1, ···, |𝚵|} do
            ζ_i ← Construct−Ξ_i(x, P, t);
        end for
        return arg max{ζ_i | i ∈ σ};
    end if
end
```

As it can be seen, the hyperheuristic firstly checks whether there is agreement among the available LLHs on which patient class to pick. If there is, no further computation is required and the unanimous decision is returned. If this is not the case, each of the associated construction heuristics is run to determine which selection is more beneficial. Note that even though two or more LLHs may agree on the choice to be made, as long as there is no unanimous decision all of them must be run since any LLH could in principle return the best function value (i.e., number of patients treated).

Notice that the two methods presented above are related since the hyperheuristic actually uses in part the philosophy of the pilot method. Indeed, Choice−Pilot(Ξ) and Choice−Hyper($\boldsymbol{\Xi}$) can be regarded as two complementary approaches: the first one can take any decision at a given time using a single heuristic Ξ as pilot; the second one can only take a limited set of decisions at any time (only those returned by the LLH set $\boldsymbol{\Xi}$), but uses multiple LLHs as independent pilots for each decision. Furthermore, it is possible to define a blended approach Pilot(Hyper($\boldsymbol{\Xi}$)), that uses the hyperheuristic with LLH set $\boldsymbol{\Xi}$ as pilot for the construction process. This approach and all preceding ones will be experimentally compared next.

4 Experimental Results

We have conducted an extensive empirical evaluation of the heuristics presented. The experimental setup is similar to that used in [20]. To be precise, we have initially considered a problem formulation involving two patient classes. These can be regarded as the two most critical classes (excluding expectant casualties) of typical field triage methods, since patients in the 'green' class are usually delayed until these most critical casualties are treated. Notice at any rate that later on we will test the scalability of heuristics in a 3-class scenario. We have generated $N = 5,000$ problem instances for each of three different severity conditions. In all cases, this first set of experiments assumes operating times τ_i are uniformly distributed in (0.5,2.0); we enforce $\tau_1 > \tau_2$, i.e., class c_1 patients require more time to be operated than those of class c_2. As to the lifetime distribution, we assume it to be Weibull-distributed with shape parameter $\alpha_i = 1.5$ (i.e., increasing hazard rate). The scale parameter β_i is set such that the corresponding initial abandonment rate r_i is within a given interval. These intervals represent different severity conditions as mentioned before. Thus, we have $r_i \in (0.1, 0.5)$ (denoted as S1), $r_i \in (0.5, 2.0)$ (S2) and $r_i \in (2.0, 5.0)$ (S3), respectively representing increasingly critical conditions (in the first case operating rates are higher than abandonment rates, in the second case they are in the same interval, and in the third case, abandonment rates are higher than operating rates). Again, we enforce $r_1 > r_2$ so that c_1 patients are more critical than c_2 patients. The initial number of patients $x_i \in [1, 20]$ in each class is uniformly selected at random in each instance, and the number of operating rooms ϱ is set to 5.

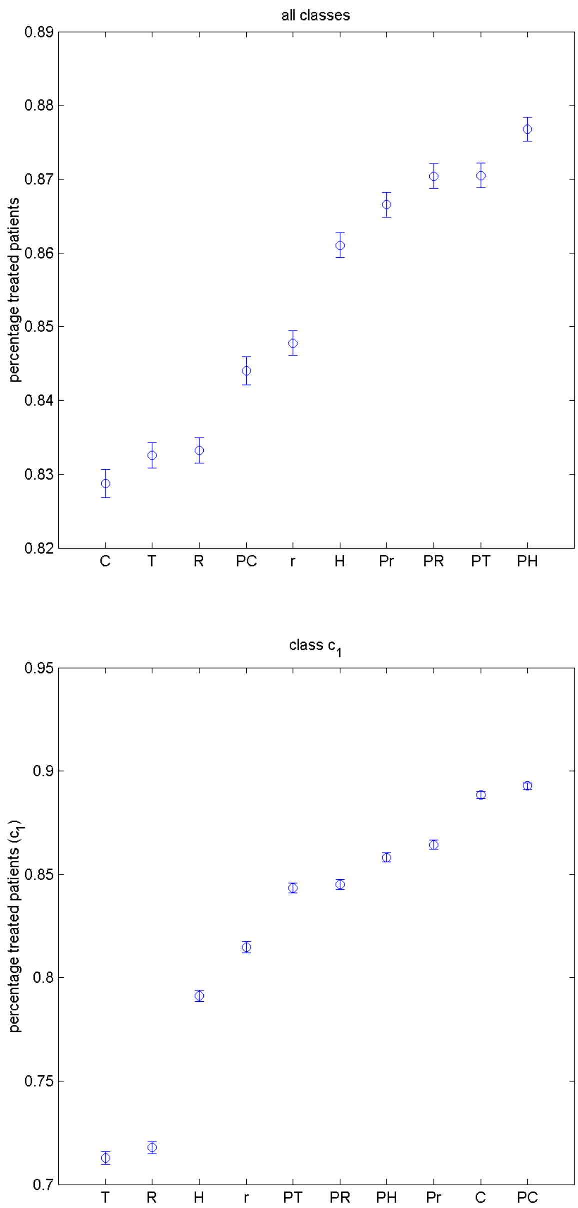

Fig. 1 Percentage of patients treated in scenario S1, using Weibull-distributed lifetimes and deterministic operating times. The top figure corresponds to the mean percentage of patients treated in both classes, and the bottom one to patients in the most critical class. The error bars indicate the standard deviation of the mean. In this figure and in all subsequent ones, algorithms are labeled as C = Time Critical First, T = Triangular, R = Rectangular, r = $r\mu$, H = Hyper(Ξ), PX = Pilot(X). Note the different ordering of algorithms in each subfigure.

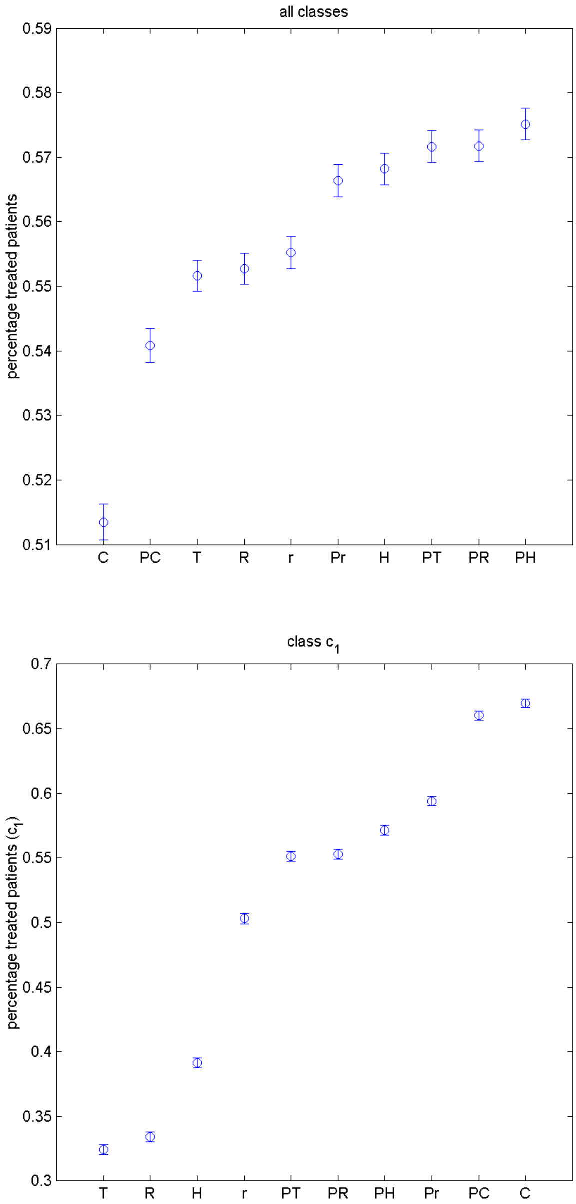

Fig. 2 Percentage of patients treated in scenario S2, using Weibull-distributed lifetimes and deterministic operating times. The top figure corresponds to the mean percentage of patients treated in both classes, and the bottom one to patients in the most critical critical class. The error bars indicate the standard deviation of the mean. Note the different ordering of algorithms in each subfigure.

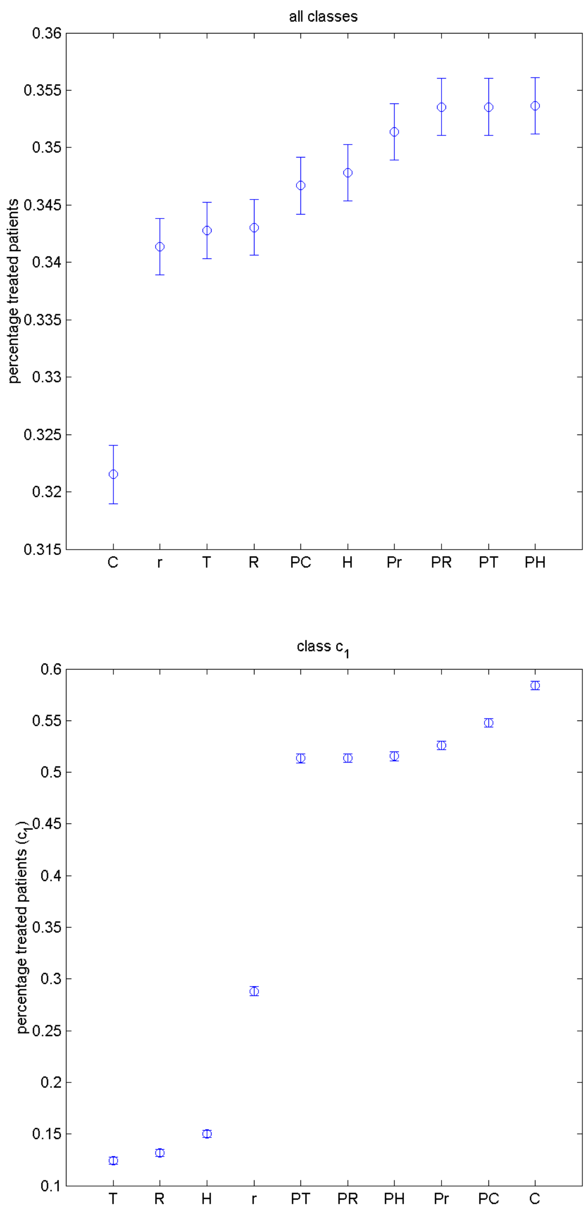

Fig. 3 Percentage of patients treated in scenario S3, using Weibull-distributed lifetimes and deterministic operating times. The top figure corresponds to the mean percentage of patients treated in both classes, and the bottom one to patients in the most critical critical class. The error bars indicate the standard deviation of the mean. Note the different ordering of algorithms in each subfigure.

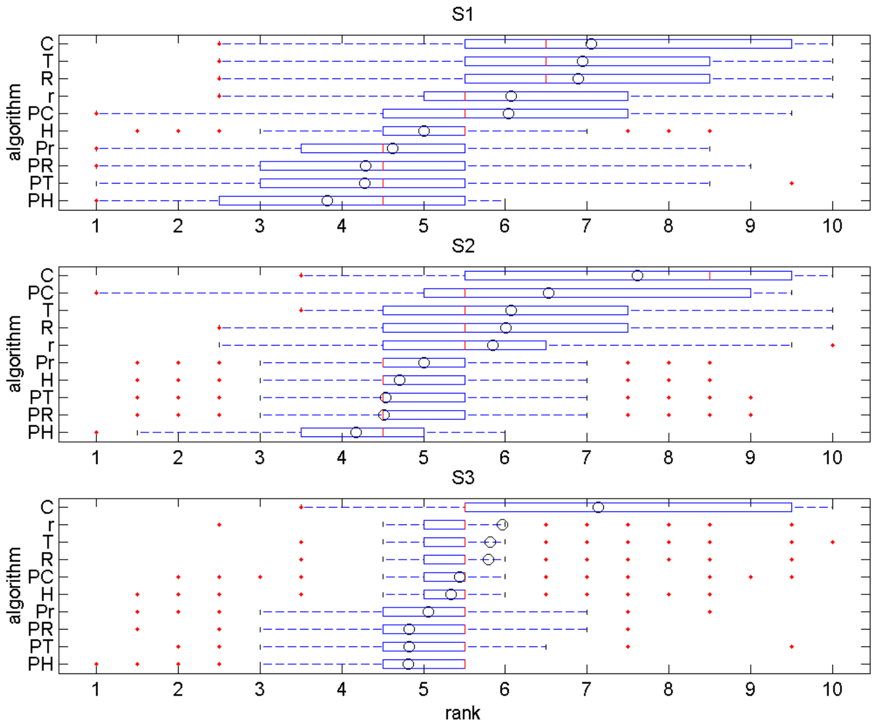

Fig. 4 Rank distribution of the different algorithms in the three scenarios considered (Weibull-distributed lifetimes, deterministic operating times) in increasing criticality from top (S1) to bottom (S3). As usual each box comprises the second and third quartiles, the vertical line marks the median, the circle marks the mean, the whiskers span 1.5 times the interquartile-distance, and the dots are outliers.

The results for the three scenarios are shown in Figures 1–3. Notice firstly the outcome of the basic heuristics T, R, $r\mu$ and TCF (respectively labelled as T, R, r, and C in the figure). Consistently with [20], T and R perform better in a scenario in which abandonment rates are very high, whereas $r\mu$ performs better in scenarios of more moderate severity. Likewise, TCF provides the worst results, in particular in the most critical scenarios in which the very myopic policy of focusing on class c_1 results in multiple impatient deaths in class c_2. Conversely, when the situation is less critical, TCF is comparatively closer to the remaining basic heuristics since less patients leave class c_2 before treatment.

Consider now the results of the Pilot(Ξ) and Hyper($\boldsymbol{\Xi}$). Regarding the former, pilot methods based on each of the four basic heuristics have been considered. As to the latter, we have considered $\boldsymbol{\Xi} = \{\text{T, R, } r\mu\}$, leaving out TCF due to its poorer performance. As it can be seen, there is a marked difference between basic heuristics and the corresponding pilot method. Notice

Table 1 Results of Holm's test using PILOT(HYPER(Ξ)) as control algorithm.

S1: $r_i \in (0.1, 0.5)$

i	algorithm	p-value	α/i
9	TCF	0	0.00556
8	T	0	0.00625
7	R	0	0.00714
6	$r\mu$	3.46e-302	0.00833
5	PILOT(TCF)	3.63e-293	0.01000
4	HYPER(Ξ)	1.32e-085	0.01250
3	PILOT($r\mu$)	3.45e-040	0.01667
2	PILOT(R)	6.10e-015	0.02500
1	PILOT(T)	1.32e-014	0.05000

S2: $r_i \in (0.5, 2.0)$

i	algorithm	p-value	α/i
9	TCF	0	0.00556
8	PILOT(TCF)	0	0.00625
7	T	1.91e-215	0.00714
6	R	2.48e-199	0.00833
5	$r\mu$	4.99e-168	0.01000
4	PILOT($r\mu$)	1.53e-042	0.01250
3	HYPER(Ξ)	2.01e-018	0.01667
2	PILOT(T)	2.78e-009	0.02500
1	PILOT(R)	8.94e-009	0.05000

S3: $r_i \in (2.0, 5.0)$

i	algorithm	p-value	α/i
9	TCF	0	0.00556
8	$r\mu$	5.62e-080	0.00625
7	T	9.31e-062	0.00714
6	R	8.11e-059	0.00833
5	PILOT(TCF)	3.34e-025	0.01000
4	HYPER(Ξ)	2.93e-018	0.01250
3	PILOT($r\mu$)	2.72e-005	0.01667
2	PILOT(R)	0.42	0.02500
1	PILOT(T)	0.42	0.05000

also that the hyperheuristic also provides better results than those of the basic heuristics. Although differences seem smaller in the most critical scenario, they are still significant. Actually, a Wilcoxon signed-rank test [37] (used to perform a statistical comparison on paired samples) indicates that in all cases both PILOT(Ξ) and HYPER(Ξ) are significantly (at the standard 0.05 level) better than the corresponding LLH Ξ. Furthermore, check Figure 4 in which we plot the distribution of ranks of each algorithm in each of the three scenarios. These ranks are computed by sorting the algorithms on each of the $5,000$ instances, assigning rank 1 to the best algorithm in a certain instance and rank k (k being the total number of algorithms) to the worst one. In case of a tie, the average of the positions involved is used as rank.

PILOT(HYPER(Ξ)) consistently provides the best rank, followed by PILOT(T) and PILOT(R) that rank close to each other (like T and R do). To ascertain the significance of these ranks, we have firstly performed both Friedman's test [38] and Iman-Davenport's test [39] on the data. Both tests indicate that there are significant differences, so we have subsequently performed Holm's test [40] using PILOT(HYPER(Ξ)) –the algorithm with the best mean rank– as control algorithm. The results of the test –shown in Table 1– indicate that this control algorithm ranks significantly better than the remaining algorithms in S1 and S2. In S3 no significant difference in rank can be found for PILOT(R), PILOT(T) and PILOT(HYPER(Ξ)). This result can be explained by the improved performance of T and R in this scenario boosting the corresponding pilot methods as well. As an aside note, computational times per problem instance were around 1-2ms milliseconds for the LLHs, about 6-40ms for pilot

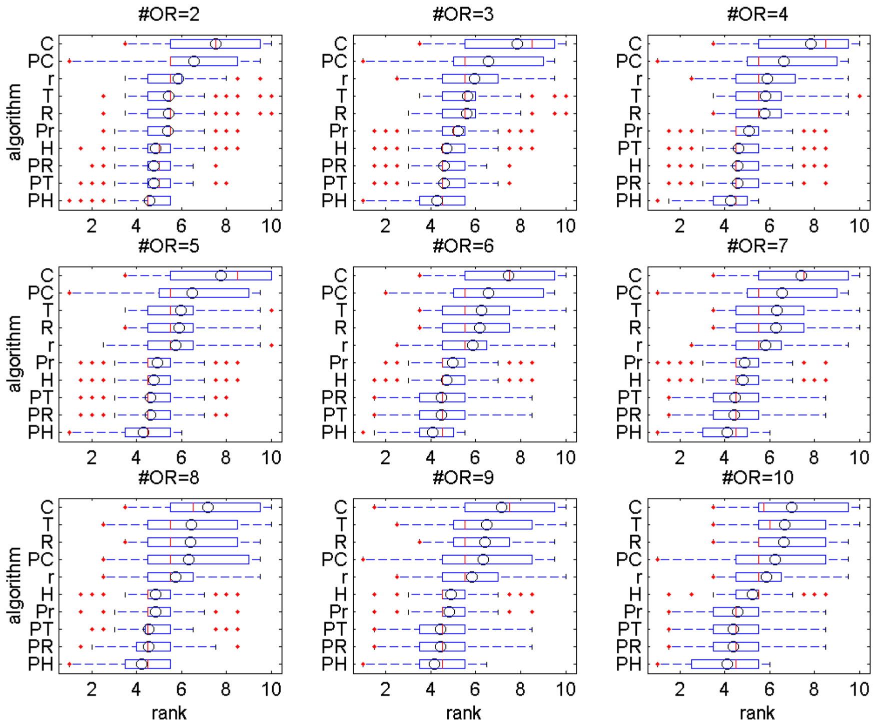

Fig. 5 Rank distribution of the different algorithms in scenario S2 as a function of the number of operating rooms available.

methods and the hyperheuristic, and about 0.3s for Pilot(Hyper(Ξ)) (times measured on an Intel Core 2 Quad Q6600 2.4 GHz).

Next, we have done experiments in order to determine the influence of some problem parameters on the performance of the different methods. In first place, we have analyzed the impact of having a different number of operating rooms available. To this end, we have again generated 5,000 instances as defined above, each of them with a number of operating rooms ϱ drawn from a uniform distribution $\varrho \in \{2, \cdots, 10\}$. Subsequently, we have grouped problem instances according to the number of operating rooms, and performed a separate rank analysis on each group. The results are consistent with those shown before for $\varrho = 5$. In the most critical scenario S3, Holm's test rejects differences between Pilot(R), Pilot(T) and Pilot(Hyper(Ξ)) regardless of the number of operating rooms. In S1 and S2, the test is passed using the latter as control algorithm. Figure 5 shows the rank distribution for S2. Note that T and R perform better than $r\mu$ for lower number of operating rooms, whereas for a larger number the opposite is true. This can be interpreted in light of the myopic measure of expected number of abandonments during operation

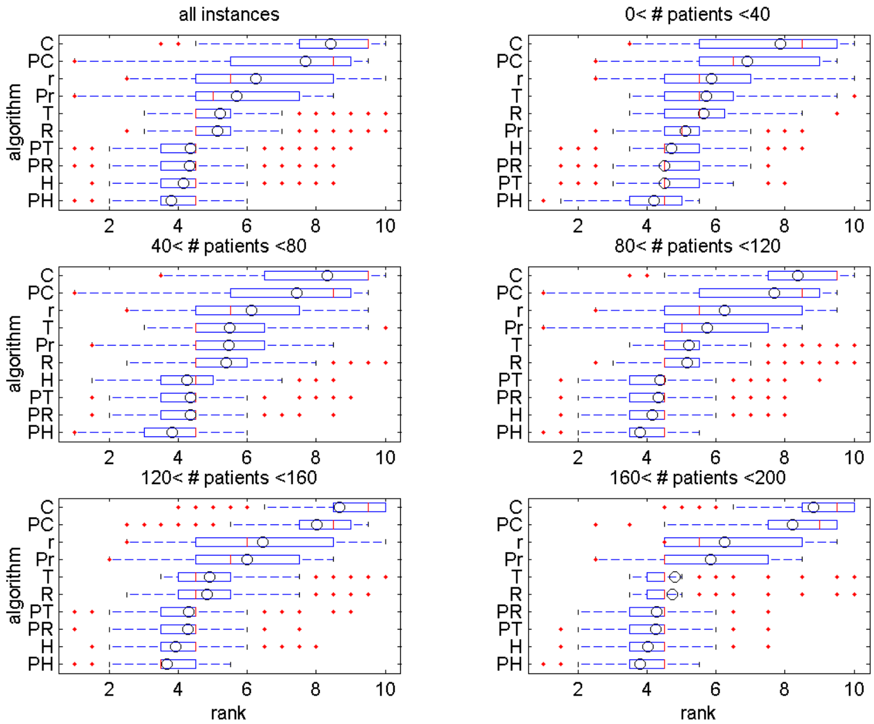

Fig. 6 Rank distribution of the different algorithms in scenario S2 as a function of the total number of patients.

not coping well with the fact that there may be many operations in parallel. More foresight is required in this case to achieve better decision-making. Note in this sense that for very low values of ϱ rank differences are lower as well (yet still statistically significant). Pilot methods keep performing the best, closely followed by the hyperheuristic, which ranks the third for $\varrho < 7$, ties with PILOT($r\mu$) for $7 \leqslant \varrho \leqslant 9$ (no statistical difference using a Wilcoxon signed-rank test), and is only overcome by the latter for $\varrho = 10$. Computational times per instance are 1-4ms for LLHs, about 15-70ms for pilot methods and the hyperheuristic, and about 0.8s for PILOT(HYPER(Ξ)).

The next issue to be tackled is the scalability of heuristics, either in terms of the number of patients or in the number of patient classes. Regarding the former we have repeated the experiments following the previous methodology but using this time an initial number of patients $x_i \in [1, 100]$, i.e., a 5-fold increase in the upper limit. In this case, T and R are found to be better than $r\mu$ for larger number of patients. See for example Figure 6, in which ranks are shown for S2. Wilcoxon signed-rank test indicates that T and R perform significantly better than $r\mu$ for more than 40 patients. Notice also

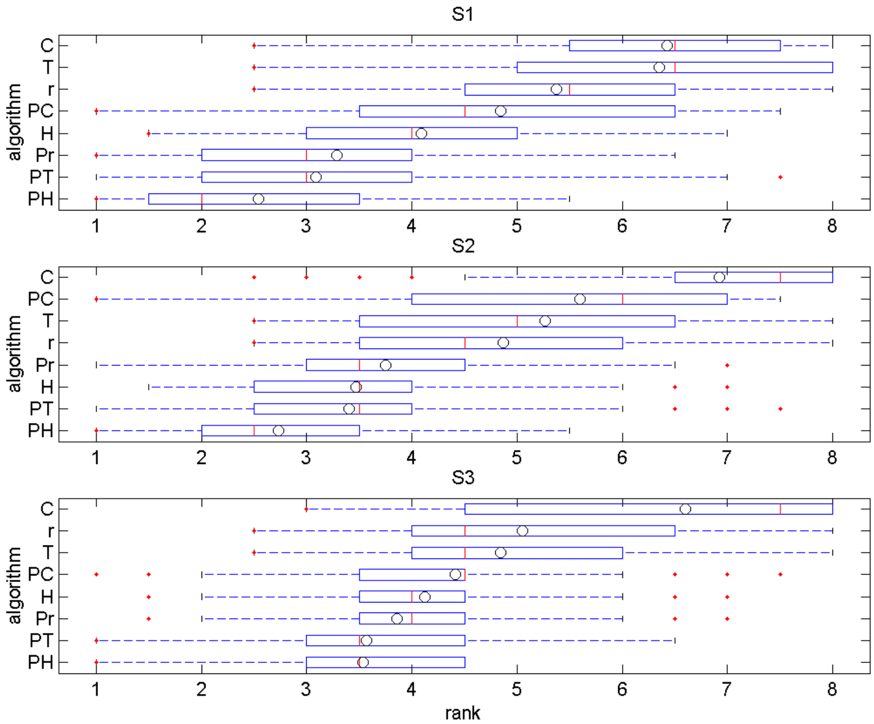

Fig. 7 Rank distribution of the different algorithms in S1 (top), S2 (middle) and S3 (bottom) when patients are sorted into three classes.

the improved performance of the Hyper(Ξ) for larger number of patients, only second after Pilot(Hyper(Ξ)). This suggests using several pilots (either directly or indirectly) in these larger instances as a more scalable strategy. Computational times per instance range in this case from 5-10ms for the LLHs to 0.1-0.3s for pilot methods and the hyperheuristic, and are about 4.7s for Pilot(Hyper(Ξ)).

Subsequently, we have considered patients classified into 3 classes rather than 2. We have generated $5,000$ instances enforcing as before that more critical patients have also longer operation times. The ranks of the algorithms are shown in Figure 7. Note that since heuristic R is not directly generalizable to more than two classes, we have used neither it nor Pilot(R) in this case (also, Hyper(Ξ) does not include R in the LLH set). The results in this case are qualitatively the same as for two classes, T outperforming $r\mu$ in S3 and Pilot(Hyper(Ξ)) being the best algorithm in S1 and S2 (Holm's test is passed). In S3, Holm's test rejects there is a significant difference between Pilot(Hyper(Ξ)) and Pilot(T), as it was the case for two classes. As to the hyperheuristic, it ranks consistently above the basic heuristics. We have

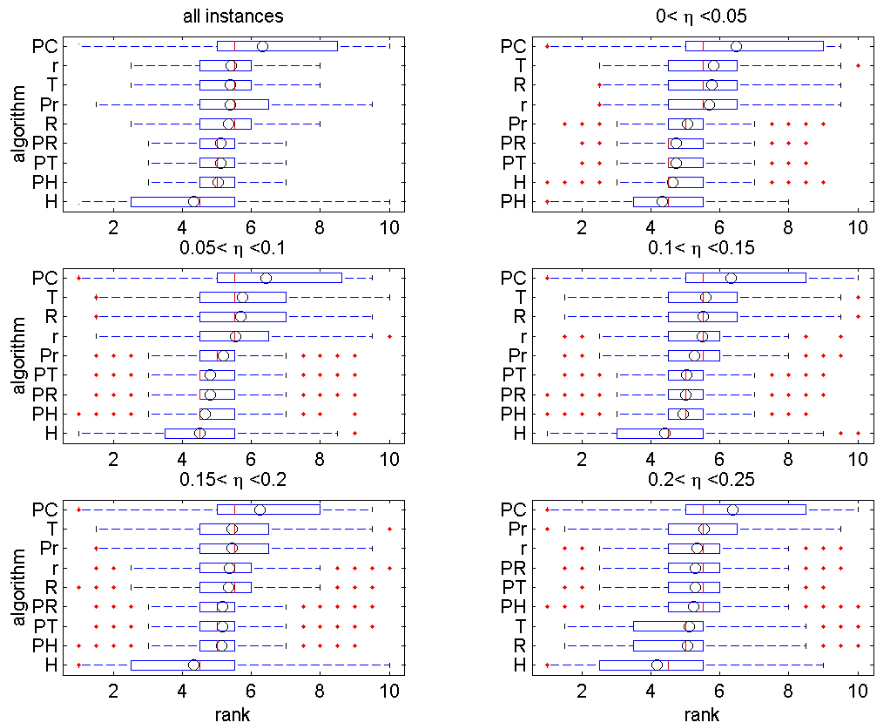

Fig. 8 Rank distribution of the different algorithms in S2 for stochastic operation times.

also conducted experiments on a mixed scenario in which the abandonment rates of c_1 patients correspond to S3 (the most critical scenario), those of c_2 patients correspond to S2, and those of c_3 to S1 (the less critical scenario). In this case the performance of T is degraded due to its myopic choice function being too conservative and resulting in many impatient deaths in c_1. This performance drop also affects the hyperheuristic, which performs comparably to PILOT(TCF) (no statistical difference according to Wilcoxon signed-rank test), and below the remaining pilot methods. Computational times per instance are in this case: 1-5ms for the LLHs, 0.02-0.16s for pilot methods and the hyperheuristic, and about 1.8s for PILOT(HYPER(Ξ)).

Finally, experiments have been done to determine the influence that stochastic operation times have on the performance of the heuristics. As mentioned in Section 3.1, we model this by assuming operating a patient in class c_i takes a minimum time τ_i plus an excess time which is exponentially distributed with parameter $1/(\eta_i \tau_i)$. We consider $\eta_i \in (0, 0.25)$, and enforce $\eta_1 > \eta_2$. In this new scenario, pilot methods use the minimum time τ_i as an optimistic approximation to operation time. We have generated $5,000$ instances for each scenario S1, S2 and S3. We perform $M = 15$ runs of each algorithm on each

Table 2 Statistical analysis of the results using Wilcoxon ranksum test. Each entry in the table indicates the percentage of instances in which the algorithm labelled in the row outperforms the algorithm labelled in the column. Note that the sum of diagonal-symmetric entries do not necessarily add up to 100% since it is possible that no statistically significant difference can be established for certain instances.

S1: $r_i \in (0.1, 0.5)$

						Pilot				
	T	R	$r\mu$	TCF	H	T	R	$r\mu$	TCF	H
T	–	0	3	28	0	6	6	7	17	6
R	1	–	3	28	0	6	6	7	17	6
$r\mu$	24	23	–	25	1	6	6	6	16	5
TCF	26	25	2	–	3	1	1	0	0	0
Hyper(Ξ)	33	33	21	46	–	12	12	15	26	9
Pilot(T)	45	44	29	46	16	–	0	4	21	0
Pilot(R)	45	44	29	46	16	0	–	4	21	0
Pilot($r\mu$)	44	44	28	44	17	4	4	–	17	1
Pilot(TCF)	40	39	24	34	13	6	5	2	–	2
Pilot(Hyper(Ξ))	47	47	34	50	18	5	5	7	24	–

S2: $r_i \in (0.5, 2.0)$

						Pilot				
	T	R	$r\mu$	TCF	H	T	R	$r\mu$	TCF	H
T	–	0	12	51	0	16	16	21	36	16
R	2	–	13	51	0	16	16	21	36	16
$r\mu$	12	11	–	40	1	12	12	13	29	12
TCF	13	12	1	–	2	0	0	0	0	0
Hyper(Ξ)	22	21	21	59	–	23	23	27	43	22
Pilot(T)	20	19	17	49	6	–	0	7	29	2
Pilot(R)	20	19	17	49	6	0	–	7	29	2
Pilot($r\mu$)	19	18	11	44	6	2	2	–	23	1
Pilot(TCF)	17	16	9	30	4	4	4	2	–	3
Pilot(Hyper(Ξ))	22	21	17	50	6	3	3	8	30	–

S3: $r_i \in (2.0, 5.0)$

						Pilot				
	T	R	$r\mu$	TCF	H	T	R	$r\mu$	TCF	H
T	–	0	8	35	0	7	7	8	10	7
R	0	–	8	35	0	7	7	8	10	7
$r\mu$	3	3	–	27	1	4	4	4	6	4
TCF	4	4	1	–	2	0	0	0	0	0
Hyper(Ξ)	5	5	10	37	–	7	7	9	11	8
Pilot(T)	11	11	13	34	7	–	0	3	8	1
Pilot(R)	11	11	13	34	7	0	–	3	8	1
Pilot($r\mu$)	11	11	13	33	7	1	1	–	5	1
Pilot(TCF)	12	12	13	32	8	4	4	3	–	4
Pilot(Hyper(Ξ))	12	12	13	34	7	0	0	3	8	–

problem instance, and take the median value for ranking purposes. Computational times per instance are 1-4ms for LLHs, about 10-70ms for pilot methods and the hyperheuristic, and about 0.9s for Pilot(Hyper(Ξ)).

As expected, the results indicate that pilot methods are sensitive to the presence of noise, although their performance do not degrade excessively. Quite interestingly, T and R heuristics improve their relative performance in S2 and

S3 with increasing values of η_i, e.g., see Figure 8. A ranksum test on T and R vs. their respective pilot methods indicate that the rank differences are significant (except for R vs. PILOT(R) in $0.15 < \max(\eta_1, \eta_2) \leqslant 0.2$). In particular, this means that when all instances are considered there is a moderate but significant advantage of the pilot methods. The hyperheuristic ranks the first (or statistically indistinguishable from the first) in S3 for $\max(\eta_1, \eta_2) \geqslant 0.1$, and in S2 for $\max(\eta_1, \eta_2) \geqslant 0.05$ (in S1 the best algorithm is PILOT(HYPER(Ξ)) as supported by Holm's test). The reason why the hyperheuristic performs better than other pilot methods in these instances can be found in the fact that the former is less 'risky': it accepts the choice taken by the LLHs if there is agreement among them; on the contrary, other pilot methods may take decisions departing from those of a certain LLH on the basis of future gains as indicated by looking ahead. However, if there is uncertainty in this information a risky decision might not be ultimately as beneficial as initially thought.

Table 2 provides a global perspective of how the different techniques compare in this last setting. Entries in the table indicate the percentage of instances in which a certain algorithm outperform another one (with statistical significance using a Wilcoxon ranksum test to compare all runs of two algorithms on each problem instance). Across the whole set of instances there is a trend of superiority for the metaheuristic approaches. This superiority is less marked in S3 where pilot methods are better than the LLHs they are based on in a net 4-5% of instances (save for TCF where the net superiority is much larger). This can be explained by the extremely severe condition of patients in this scenario, whose abandonment rates are larger than operating rates, leading to many impatient deaths in all cases (thus leaving a narrow margin for improvement). The uncertainties in operation times are also large with regard to survival expectancies in that scenario, hence the higher impact they have on pilot methods. On the other hand, the metaheuristics are remarkably better than the LLHs in S1, where the uncertainty in operation times is comparatively smaller with respect to the larger life expectancy of patients.

5 Conclusions

The allocation of limited medical resources to the victims of a mass casualty incident is a complex task due to the number of factors involved (not to mention its ethical ramifications). The main contribution of this paper has been the design and extensive analysis of metaheuristics (hyperheuristics and the pilot method) for dealing with this decision-making problem. The results have been positive and provide evidence on the potential usefulness of this kind of methods in this context. Quoting [41], the science of triage (in particular, tertiary triage, namely the effective assignment of limited resources under competing patient demands) is nascent, and very much in need of more robust and researched strategies. In this sense, the techniques described in this work should be considered as a step in this direction. Indeed, metaheuristic approaches have been shown as effective high-level methods to coordinate the application of low level heuristics designed for this prioritization problem. The results indicate they are competitive in multiple scenarios with different features regarding the number of operating rooms, patients, and triage classes. They are however sensitive to the presence of large uncertainties in operation times, in particular in the most severe scenarios where these uncertainties are comparatively larger with respect to lifetime expectancies. It is possible to conceive the use of specialized mechanisms to deal with uncertainty in this context. This constitutes a line of future work.

There are many other avenues for further research. Regarding the problem model, more complex scenarios could be considered, e.g., involving survival probabilities after treatment. Other metaheuristic frameworks, e.g., evolutionary algorithms, could be used here as well. The use of population-based techniques would involve among other issues investigating whether they can provide an adequate tradeoff between performance and computation-time. The application of machine learning strategies is also worth considering as a means to adaptively control the application of low level heuristics in this domain.

Acknowledgements The author wishes to thank the referees for their useful comments which have contributed to improve the paper.

References

1. Koenig, K.L., Dinerman, N., Kuehl, A.E.: Disaster nomenclature – a functional impact approach: The PICE system. Academic Emergency Medicine **3**(7) (1996) 723–727
2. van Veelen, J., van Aart, C., Storms, P.: Effective and efficient coordination strategies for agile crisis response organizations. In de Walle, B.V., Turoff, M., eds.: Proceedings of the 3rd International ISCRAM Conference, Newark, NJ (2006) 202–213
3. Doerner, K.F., Hartl, R.F.: Health care logistics, emergency preparedness, and disaster relief: New challenges for routing problems with a focus on the Austrian situation. In Golden, B., et al., eds.: The Vehicle Routing Problem : Latest Advances and New Challenges. Volume 43 of Operations Research/Computer Science Interfaces Series. Springer-Verlag, Berlin (2008) 527–550
4. Tatomir, B., Rothkrantz, L.: Ant based mechanism for crisis response coordination. In Dorigo, M., et al., eds.: Ant Colony Optimization and Swarm Intelligence. Volume 4150 of Lecture Notes in Computer Science., Brussels, Springer-Verlag (2006) 380–387
5. Lee, E.K., et al.: Facility location and multi-modality mass dispensing strategies and emergency response for biodefence and infectious disease outbreaks. International Journal of Risk Assessment and Management **12** (2009) 311–351
6. Frykberg, E.: Principles of mass casualty management following terrorist disasters. Annals of Surgery **239**(3) (2004) 319–321

7. Iserson, K.V., Moskop, J.C.: Triage in medicine, part I: Concept, history, and types. Annals of Emergency Medicine **49**(3) (2007) 275–281
8. Kennedy, K., et al.: Triage: Techniques and applications in decisionmaking. Annals of Emergency Medicine **28**(2) (1996) 136–144
9. Robertson-Steel, I.: Evolution of triage systems. Emergency Medicine Journal **23**(2) (2006) 154–155
10. Bostick, N., et al.: Disaster triage systems for large-scale catastrophic events. Disaster Medicine and Public Health Preparedness **2**(Suppl. 1) (2008) S35–S39
11. Bar-Joseph, G., Michaelson, M., Halberthal, M.: Managing mass casualties. Current Opinion in Anaesthesiology **16** (2003) 193–199
12. Roccaforte, J., Cushman, J.: Disaster preparedness, triage, and surge capacity for hospital definitive care areas: Optimizing outcomes when demands exceed resources. Anesthesiology Clinics **25**(1) (2007) 161–177
13. Cowling, P.I., Kendall, G., Soubeiga, E.: A hyperheuristic approach to scheduling a sales summit. In Burke, E.K., Erben, W., eds.: Practice and Theory of Automated Timetabling III. Volume 2079 of Lecture Notes in Computer Science., Konstanz, Germany, Springer-Verlag (2000) 176–190
14. Burke, E., et al.: Hyper-heuristics: an emerging direction in modern search technology. In Glover, F., Kochenberger, G., eds.: Handbook of Metaheuristics. Volume 57 of International Series in Operations Research and Management Science. Kluwer Academic Pub. (2003) 457–474
15. Chakhlevitch, K., Cowling, P.I.: Hyperheuristics: Recent developments. In Cotta, C., Sevaux, M., Sörensen, K., eds.: Adaptive and Multilevel Metaheuristics. Volume 136 of Studies in Computational Intelligence. Springer-Verlag (2008) 3–29
16. Özcan, E., Bilgin, B., Korkmaz, E.E.: A comprehensive analysis of hyperheuristics. Intelligent Data Analysis **12**(1) (2008) 3–23
17. Duin, C., Voß, S.: Steiner tree heuristics – A survey. In Dyckhoff, H., et al., eds.: Operation Research Proc 1993 (DGOR-NSOR), Berlin, Springer-Verlag (1994) 485–496
18. Duin, C., Voß, S.: The pilot method: A strategy for heuristic repetition with application to the Steiner problem in graphs. Networks **34** (1999) 181–191
19. Voß, S., Fink, A., Duin, C.: Looking ahead with the pilot method. Annals of Operations Research **136**(1) (2005) 285–302
20. Argon, N.T., Ziya, S., Righter, R.: Scheduling impatient jobs in a clearing system with insights on patient triage in mass casualty incidents. Probability in the Engineering and Informational Sciences **22**(3) (2008) 301–332
21. Rubinson, L., et al.: Augmentation of hospital critical care capacity after bioterrorist attacks or epidemics: recommendations of the working group on emergency mass critical care. Critical Care Medicine **33** (2005) 2393–2403
22. Hick, J., Barbera, J., Kelen, G.: Refining surge capacity: Conventional, contingency, and crisis capacity. Disaster Medicine and Public Health Preparedness **3**(Suppl 1) (2009) S59–S67

23. Challen, K., et al.: Clinical review: Mass casualty triage - pandemic influenza and critical care. Critical Care **11** (2007) 212–216
24. Daniels, N.: Accountability for reasonableness. British Medical Journal **321** (2000) 1300–1301
25. Good, L.: Ethical decision making in disaster triage. Journal of Emergency Nursing **34**(2) (2008) 112–115
26. Larkin, G., Arnold, J.: Ethical considerations in emergency planning, preparedness, and response to acts of terrorism. Prehospital Disaster Medicine **18**(3) (2003) 170–178
27. Cone, D.C., MacMillan, D.S.: Mass-casualty triage systems: A hint of science. Academic Emergency Medicine **12**(8) (2005) 739–741
28. Super, G.: START: A Triage Training Module. Hoag Memorial Presbyterian, Newport Beach, CA (1984)
29. Garner, A., et al.: Comparative analysis of multiple-casualty incident triage algorithms. Annals of Emergency Medicine **38**(5) (2001) 541–548
30. Hirshberg, A.: Multiple casualty incidents. Lessons from the front line. Annals of Surgery **239**(3) (2004) 322–324
31. Stein, M., Hirshberg, A.: Medical consequences of terrorism. The conventional weapon threat. Surgical Clinics of North America **79** (1999) 1537–1552
32. Peleg, K., et al.: Gunshot and explosion injuries: Characteristics, outcomes, and implications for care of terror-related injuries in Israel. Annals of Surgery **239**(3) (2004) 311–318
33. Saffle, J., Gibran, N., Jordan, M.: Defining the ratio of outcomes to resources for triage of burn patients in mass casualties. Journal of Burn Care and Rehabilitation **26** (2005) 478–482
34. Weibull, W.: A statistical distribution function of wide applicability. Journal of Applied Mechanics **18**(3) (1951) 293–297
35. Glazebrook, K., et al.: On the optimal allocation of service to impatient tasks. Journal of Applied Probability **41**(1) (2004) 51–72
36. Lee, E.T., Wang, J.W.: Statistical Methods for Survival Data Analysis. John Wiley & Sons, Inc., Hoboken, NJ (2003)
37. Wilcoxon, F.: Individual comparisons by ranking methods. Biometrics **1** (1945) 80–83
38. Friedman, M.: The use of ranks to avoid the assumption of normality implicit in the analysis of variance. Journal of the American Statistical Association **32**(200) (1937) 675–701
39. Iman, R., Davenport, J.: Approximations of the critical region of the Friedman statistic. Communications in Statistics **9** (1980) 571–595
40. Holm, S.: A simple sequentially rejective multiple test procedure. Scandinavian Journal of Statistics **6** (1979) 65–70
41. Hick, J., et al.: Clinical review: Allocating ventilators during large-scale disasters - problems, planning, and process. Critical Care **11**(3) (2007) 217–225