

# Clustering Clinical Departments for Wards to Achieve a Prespecified Blocking Probability

J. Theresia van Essen<sup>\*1</sup>, Mark van Houdenhoven<sup>2</sup>, Johann L. Hurink<sup>1</sup>

<sup>1</sup>Center for Healthcare Operations Improvement and Research (CHOIR),  
University of Twente, Enschede, The Netherlands  
<sup>2</sup>HagaZiekenhuis, Den Haag, The Netherlands

## Abstract

When the number of available beds in a hospital is limited and fixed, it can be beneficial to cluster several clinical departments such that the probability of not being able to admit a patient is acceptably small. The clusters are then assigned to the available wards such that enough beds are available to guarantee a blocking probability below a prespecified value. We first give an exact formulation of the problem to be able to achieve optimal solutions. To reduce computation times, we also introduce two heuristic solution methods. The first heuristic is similar to the exact solution method, however, the number of beds needed is approximated by a linear function. The second heuristic uses a local search approach to determine the assignment of clinical departments to clusters and a restricted version of the exact solution method to determine the assignment of clusters to wards.

**Keywords:** Hospitals, Lay-out, Wards, Integer programming, Heuristics

## Introduction

The HagaZiekenhuis, a hospital in the Netherlands, is forced by health insurance companies to reduce the number of beds available in the hospital with the goal to reduce the health care costs. Therefore, the hospital has the challenging task to treat the same amount of patients with less beds available. As the demand for care will increase in the future due to the ageing population and more chronic ill patients, the hospital expects that it even has to be able to treat more patients with this reduced bed capacity. The considered problem does not only occur in the HagaZiekenhuis, but also in other hospitals in the Netherlands, and most likely, also in hospitals in other countries.

Treating the same amount of patients or even more with a reduced number of beds can be achieved in different ways. First, if the length of stay (LOS) of patients is reduced, the bed capacity needed to treat the same amount of patients decreases. However, it might not always be possible to reduce the LOS, e.g., due to the specific illness of patients. A second way to reduce the bed capacity is to level the bed occupancy over time. This can, for example, be done by adjusting the operating room schedule (van Essen et al., 2012). Third, by clustering several (small) clinical departments (i.e. by assigning several clinical departments to the same wards such that the departments share the beds available at these wards), the bed capacity can also be reduced. The

---

<sup>\*</sup>j.t.vanessen@utwente.nl

basic reason for this is that when two or more clinical departments are clustered, also the risk of refusing a patient is reduced which leads to less beds needed in total.

In this paper, we consider the third option to solve the problem of HagaZiekenhuis. Thus, we aim to assign one or more clinical departments to a cluster such that the number of beds needed in total is reduced. While determining the clusters, we consider the medical limitations meaning that, for example, the general surgery and internal medicine department cannot be clustered. Each formed cluster is then assigned to one or more wards such that enough beds are available on these assigned wards to guarantee that the probability of refusing a patient lies below a given threshold. Note that not more than one cluster can be assigned to a ward to ensure that only the clinical departments assigned to the considered cluster share the beds available at these wards.

Several papers discuss methods to determine the number of beds needed for a specific clinical department such that almost all patients can be admitted. Lapierre et al. (Lapierre et al., 1999) consider the problem of assigning a number of beds to clinical departments while taking into account that the admission schedule varies over the week and that the demand can show seasonality. They develop a time series model that predicts when the demand exceeds the available number of beds. This model can help in making good decisions regarding the size of a ward.

Green and Nguyen (Green and Nguyen, 2001) use an  $M/M/s$  queueing model to estimate the delay of admission when a number  $s$  of beds is given. By considering several values for  $s$ , they can choose the number of beds such that the resulting delay of admission is acceptable for hospital management. Harper and Shahani (Harper and Shahani, 2002) present a detailed simulation model which determines the refusal rate for a given bed capacity and planning policy.

Gorunescu et al. (Gorunescu et al., 2002a) use an Erlang loss model to determine the number of beds needed by a clinical department. Hereby, they make a trade-off between the loss probability and the costs of empty beds. Gorunescu et al. (Gorunescu et al., 2002b) extend this approach by introducing a waiting room with extra beds and show that this waiting room improves the performance of the system.

Utley et al. (Utley et al., 2003) use a generating function to represent the probability distribution of the bed occupancy for each day of a given planning horizon. This probability distribution includes the scheduled arrival of elective patients while considering a no-show probability and the unscheduled arrival of emergency patients. An unlimited bed capacity is assumed such that the probability distribution represents the bed occupancy when none of the patients is blocked. With the use of this probability distribution, the number of beds needed and the corresponding blocking probability can be determined.

Kokangul (Kokangul, 2008) models the bed occupancy as the difference between two renewal processes, namely the arrival and departure process. A simulation of the bed occupancy process is used to investigate the relationship between the optimal number of beds and the bed occupancy, number of admissions and service level. The results of this simulation are used to determine mathematical relationships between these factors. These relationships are then incorporated in a nonlinear mathematical model which is used to determine an optimal number of beds for each clinical department.

Li et al. (Li et al., 2009) developed a goal programming approach to make a trade-off between the number of beds required to achieve a targeted acceptance probability and the number of beds needed to optimize daily profits. The number of beds needed is determined by means of the Erlang loss formula.

De Bruin et al. (de Bruin et al., 2010) developed a decision support system based on the Erlang loss model to determine the appropriate size of wards. De Bruin et al. (de Bruin et al., 2010) and Green and Nguyen (Green and Nguyen, 2001) also discuss the benefits of merging departments. Green and Nguyen (Green and Nguyen, 2001) show that consolidating clinical departments reduces the number of beds needed and increases the total bed occupancy rate, especially when several small clinical departments are clustered. De Bruin et al.

(de Bruin et al., 2010) show that merging clinical departments results in an improved blocking probability and bed occupancy rate as larger departments are formed.

As in several of the mentioned papers, we also use the Erlang loss formula to determine the number of beds needed for each cluster of clinical departments while considering the blocking probability chosen by the hospital. We incorporate the Erlang loss formula into a model that determines which clinical departments should be clustered and that also assigns the clusters to one or more wards. The assignment of clusters to wards is done in such a way that enough beds are available at the assigned wards to guarantee a given blocking probability. While defining the clusters, we consider the medical limitations as mentioned before, but we also try to limit the number of clinical departments assigned to one cluster. When assigning the clusters to wards, we consider the preference of clinical departments for a specific ward and the distance between the wards assigned to a specific cluster. More detailed information about the chosen modelling approach can be found in Section . In Section , an exact solution approach for this problem is discussed. As within the practical setting of HagaZiekenhuis not always precise knowledge is available on the input parameters needed for the model or these parameters might change over time, we have to be able to evaluate several what-if scenarios in order to present a robust solution. Furthermore, since the computation time for the exact solution approach is quite long, we introduce two heuristic solution methods to reduce the computation time. The first heuristic solution approach approximates the Erlang loss formula leading to a simpler model needing less computational time. The second heuristic solution approach solves the problem in two phases. The first phase clusters the clinical departments and the second phase assigns the clusters to wards. By splitting the problem up in two phases, the overall computation time is reduced. In Section , all three solution methods are tested on instances based on the situation at HagaZiekenhuis and compared according to the computation time and solution quality. In addition, we demonstrate the potential of the solution methods to investigate several scenarios and to evaluate the robustness of a given solution. Section presents conclusions and gives recommendations for further research.

Summarizing, the contribution of this paper is as follows. First, we introduce a problem, which in this combination has not been considered before in literature. Many papers already discussed clustering clinical departments and using the Erlang loss model to determine the number of beds needed, however, none of the mentioned papers investigate which clinical departments should be clustered such that all departments have enough beds available on the assigned wards. Second, we provide an exact solution method and two heuristic solution methods. The latter are needed to quickly solve this problem such that several scenarios can be evaluated.

## Problem Formulation

Each hospital consists of several clinical departments, such as general surgery, neurosurgery, orthopedics, internal medicine and cardiology. These clinical departments need a certain amount of beds to admit and treat all their patients. The needed number of beds depends on the average length of stay (LOS)  $\mu$  of patients and the expected number of admissions per day  $\lambda$ . The values for  $\mu$  and  $\lambda$  are in general different for each clinical department. Some departments such as general surgery and orthopedics admit a high number of patients per day which each have a relative short LOS (e.g. three days). The average LOS for internal medicine is often much higher (approximately 6 days). If the average LOS  $\mu$  and the number of admissions  $\lambda$  would be deterministic, i.e., each day exactly  $\lambda$  patients arrive with LOS  $\mu$ , then the corresponding clinical departments needs  $\lambda\mu$  beds to be able to admit all patients. However, in practice  $\mu$  and  $\lambda$  are stochastic, since the LOS varies per patient and the number of admissions varies per day for each clinical department. To make sure that enough beds are available in a stochastic setting, we surely need more beds than  $\lambda\mu$  to deal with the fluctuating number of admitted

patients. However, in practice, it will not be possible to always admit all patients, because then the number of beds needed has to be extremely large and a large portion of the beds will be empty for most of the time, and thus, the available capacity is not used efficiently. Therefore, the goal is to be able to admit the patients with a high probability.

Using the Erlang loss model, we can determine the probability of not being able to admit a patient, also known as the blocking probability, when  $\mu$ ,  $\lambda$  and the number of available beds  $x$  is given. The resulting blocking probability is given by (see e.g. (Tijms, 2003)):

$$P_{\text{loss}}(\lambda\mu, x) = \frac{(\lambda\mu)^x / x!}{\sum_{k=0}^x (\lambda\mu)^k / k!}. \quad (1)$$

However, for the problem considered in this paper, we would like to determine the number of beds needed when  $\mu$ ,  $\lambda$  and the blocking probability  $B$  are given. The values for  $\mu$  and  $\lambda$  can be determined from historical data provided by the hospital and the hospital itself can choose an acceptable value for the blocking probability  $B$ . When  $\mu$ ,  $\lambda$  and  $B$  are given, the goal is to find the smallest number of beds  $x$  for which  $P_{\text{loss}}(\lambda\mu, x)$  is smaller than or equal to  $B$ . This can either be done by complete enumeration or e.g. by a bisection method (Qiao and Qiao, 1998).

From equation (1) it also can be derived that clustering clinical departments can reduce the number of beds needed in total. Consider for example two clinical departments with  $\mu_1 = 7.1$ ,  $\lambda_1 = 9$  and  $\mu_2 = 6.6$ ,  $\lambda_2 = 3$  respectively. When both departments have their own ward and the blocking probability is set to 0.05, then the first department would need 70 beds and the second department would need 25 beds, which leads to 95 beds in total. But when both departments share a ward,  $\mu$  and  $\lambda$  become  $\frac{\mu_1\lambda_1 + \mu_2\lambda_2}{\lambda_1 + \lambda_2} = 6.975$  and  $\lambda_1 + \lambda_2 = 12$  respectively for the formed cluster, which leads to 89 beds in total. Thus, for this small example the number of beds can be reduced by six by clustering both departments. For the problem considered in this paper the number of beds available at the wards is limited, and thus, clustering departments might be needed to ensure that the number of beds assigned to a cluster is enough to guarantee a given blocking probability.

The problem considered in this paper can be defined as follows. Let  $D$  be the set of clinical departments and let  $W$  be the set of available wards. For each clinical department  $d \in D$  the average LOS is given by  $\mu_d$  and the expected number of admissions per day by  $\lambda_d$ . The number of available beds at ward  $w \in W$  is given by  $b_w$  and is assumed to be fixed. The clinical departments must be assigned to the wards such that enough beds are available at the wards to admit patients of the assigned clinical department with a probability  $1 - B$ . Because in practice not all clinical departments can have their own ward, we first form clusters that consist of one or more clinical departments. Note, that each clinical department is assigned to exactly one cluster such that the clusters form a partition of the clinical departments (see Figure 1). Next, the clusters are assigned to one or more wards such that the resulting blocking probability is below  $B$  for each cluster. Note, that not more than one cluster can be assigned to a ward to ensure that the beds on this ward are only shared by the clinical departments assigned to this cluster (see Figure 1). Clustering departments is not for free, since it may complicate the process of doing rounds as the patients might be scattered over multiple wards. Therefore, we have to ensure that the clinical departments are clustered as less as possible.

The assignment of clusters to wards can be considered as a bin packing problem where the bins have varying sizes. Therefore, the number of beds available on the assigned wards will in most cases be larger than the number of beds needed by the assigned cluster which can be considered as slack.

To create a feasible solution, each clinical department  $d \in D$  has to be assigned to exactly one cluster  $c \in C$ , where  $C$  is the set of clusters. Note, that the number of clusters is not prespecified, and thus, the

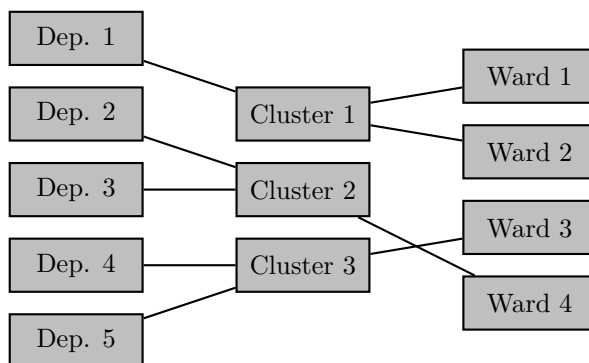


Figure 1: Example of a feasible assignment

cardinality of  $C$  is part of the decision process. To incorporate this we choose  $C$  to be sufficiently large, i.e., equal to  $\max\{|D|, |W|\}$ , but allow some clusters  $c \in C$  to be empty after the assignment process. To model the assignment of clinical departments to clusters, we introduce binary variables  $X_{dc}$  which are one when clinical department  $d \in D$  is assigned to cluster  $c \in C$ , and zero otherwise. To ensure that each clinical department  $d \in D$  is assigned to exactly one cluster  $c \in C$ , we introduce the following constraint:

$$\sum_{c \in C} X_{dc} = 1, \forall d \in D \quad (2)$$

In practice, not all clinical departments are allowed to share a ward due to medical reasons. For example, patients who had surgery and thus, have a wound that needs to heal, cannot be placed at the same ward as patients who have an infectious disease. These limitations are specified by the hospital and are modeled by variables  $a_{de}$  which are one when clinical department  $d \in D$  is allowed to be in the same cluster as clinical department  $e \in D$ , and zero otherwise. To ensure that clinical departments are only clustered when this is allowed, the following constraint is imposed:

$$X_{dc} + X_{ec} - 1 \leq a_{de}, \forall c \in C \forall d, e \in D. \quad (3)$$

The generated clusters must be assigned to one or more wards such that enough beds are available for the clusters. For this, binary variables  $Y_{cw}$  are introduced, which are one when cluster  $c \in C$  is assigned to ward  $w \in W$ , and zero otherwise. As stated before, at most one cluster may be assigned to a ward, which is ensured by the following constraint:

$$\sum_{c \in C} Y_{cw} \leq 1, \forall w \in W. \quad (4)$$

The number of required beds  $V_c$  for a cluster  $c \in C$  can be determined by the Erlang loss formula (equation (1)). Note that this number  $V_c$  cannot be determined straightforward, and thus, cannot be expressed in linear constraints. Assuming this number to be known, we have to ensure that the total number of beds available at the assigned wards is greater than or equal to  $V_c$ . This is ensured by the following constraint:

$$V_c \leq \sum_w Y_{cw} b_w, \forall c \in C. \quad (5)$$

The quality of a solution depends on several aspects. Therefore, the chosen objective function consists of three parts. First, we do not want to cluster more clinical departments into one cluster than necessary to keep

the processes at the wards simple and clear. Therefore, we aim to minimize the maximum amount of clinical departments in one cluster. The second part of our objective function is to assign clusters to wards that are close to each other. For this, we denote for each pair of wards  $w, v \in W$  the distance between those two wards by  $g_{wv}$  and add a term to the objective function that minimizes the sum of all pairwise distances of wards assigned to the same cluster. Third, some clinical departments may have a preference for a certain ward because of its location. E.g. some clinical departments, such as general surgery, orthopedics and neurosurgery, want to be close to the operating theatre. Other clinical departments, such as cardiology and thoracic surgery, want to be close to the intensive care unit. These preferences of clinical department  $d \in D$  for wards  $w \in W$  are denoted by  $h_{dw}$ . Combining these three objectives, the resulting objective function is:

$$\min \left( \max_c \sum_{d \in D} X_{dc} + \sum_{c \in C} \sum_{w \in W} \sum_{v \in W} g_{wv} Y_{cw} Y_{cv} - \sum_{c \in C} \sum_{d \in D} \sum_{w \in W} h_{dw} X_{dc} Y_{cw} \right) \quad (6)$$

Note that the problem can easily be divided into two phases. In the first phase, clinical departments are clustered and in the second phase, the clusters are assigned to wards. However, even when we assume that the assignment of clusters to wards is already fixed, the resulting problem is strongly NP-hard. To see this, assume that there are  $t$  clusters and for each cluster  $b$  beds are available on the assigned wards. Next, assume  $3t$  clinical departments, where clinical department  $d \in D$  needs  $a_d$  beds and  $\sum_{d=1}^{3t} a_d = tb$ . For this instance, determining a feasible assignment of clinical departments to wards is equivalent to determining whether there are  $t$  disjoint subsets  $R_l \subset \{1, \dots, 3t\}$  such that  $\sum_{d \in R_l} a_d = b$  for  $l = 1, \dots, t$ , which is known as the 3-partition problem (Garey and Johnson, 1979). With a similar argument we can prove that the problem is also strongly NP-hard when the assignment of clinical departments to clusters is fixed. This means that the problem consists of a combination of two strongly NP-hard problems.

## Solution Methods

For the presented problem, the average LOS  $\mu$  and the expected number of admissions per day  $\lambda$  are important input parameters. However, these parameters are both very uncertain and may change over the years: the average LOS is likely to reduce as surgeries become less invasive and diagnoses are made quicker, and due to the ageing population, the expected number of admissions per day is likely to increase as more and more people need care. This has as a consequence that in practice, several what-if scenarios have to be evaluated and, thus, fast solution methods for the stated problem are required. In Subsection , we propose such fast heuristic solution methods. Before that, in Subsection , we discuss an exact solution method. This method is used to determine optimal solutions, which are used to evaluate the performances of the heuristic solution methods.

### Exact Solution Method

The basis of the exact solution method is the formulation given in the previous section. This formulation is used to set up an Integer Linear Program (ILP) to solve the problem. The main difficulty of formulating the given problem as an ILP is determining  $V_c$ , i.e., the number of beds needed for a certain cluster  $c \in C$ . As we are not aware of an exact method to determine  $V_c$  by means of linear equations, we need to derive another method to incorporate this element. Note, that when the blocking probability  $B$  is given, the number of beds needed for a certain cluster is completely determined by  $\lambda\mu$ , which represents the expected number of admitted patients per day. Therefore, we first determine the number of beds needed for each possible value of  $\lambda\mu$  for the given problem. As the number of beds needed can only be integer, the resulting function  $f_B(\lambda\mu)$  is a stepwise

function. In Figure 2, an example is shown with  $B$  equal to 0.05 and 0.1.

From the two functions in Figure 2 it can be seen that more beds are needed when the blocking probability decreases. This should be the case, since when the blocking probability decreases, we need a larger buffer of beds to guarantee this blocking probability. When solving the problem in practice, the value of  $B$  is chosen in advance by the hospital. Therefore, the function  $f_B(\lambda\mu)$  to be used can be predetermined. It remains to determine the number of beds  $V_c$  needed for each cluster  $c \in C$ . To determine this number, we first have to determine the product of the values  $\lambda$  and  $\mu$  for each cluster  $c \in C$ . This value  $(\lambda\mu)_c$  is calculated by:

$$(\lambda\mu)_c = \sum_d X_{dc} \lambda_d \mu_d, \forall c \in C. \quad (7)$$

Then, the number of beds  $V_c$  needed for cluster  $c \in C$  can be determined as follows:

$$V_c = f_B((\lambda\mu)_c), \forall c \in C. \quad (8)$$

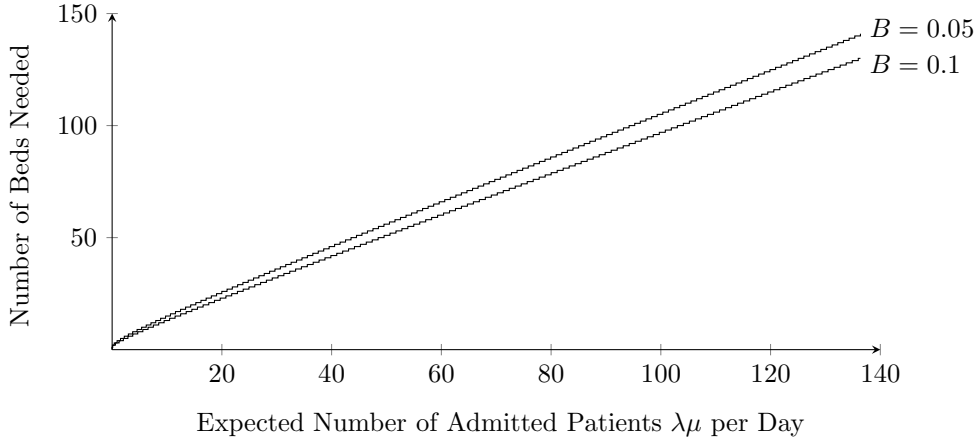


Figure 2: Exact stepwise function

As the function  $f_B(\lambda\mu)$  used in constraint (8) is not linear but a stepwise function, we have to use standard integer linear programming techniques to incorporate constraints (7) and (8) into the ILP (see e.g. (Williams, 1999)). For each cluster  $c \in C$ , we need to determine the interval of the stepwise function  $f_B(\lambda\mu)$  in which  $(\lambda\mu)_c$  lies. This leads to an extra binary variable for each cluster and interval pair. To determine this interval, we need two constraints for each cluster. The first constraint, which combines constraints (7) and (8), determines the correct interval of the stepwise function  $f_B(\lambda\mu)$ . The second ensures that only one interval is selected. Furthermore, we need a constraint similar to constraint (5) to ensure that enough beds are available on the wards assigned to cluster  $c \in C$  to guarantee the loss probability  $B$ .

Another difficulty of the formulation described in Section is the quadratic part of the objective function (6). However, this easily can be linearized by introducing binary variables  $Z_{dcw}$  and  $T_{cvw}$ . Binary variables  $Z_{dcw}$  are set to one when department  $d \in D$  is assigned to cluster  $c \in C$  which on its part is assigned to ward  $w \in W$ , and zero otherwise. The correct values of  $Z_{dcw}$  are determined by standard integer linear programming tricks which lead to the following three constraints:

$$\begin{aligned}
Z_{dcw} &\leq X_{dc}, \quad \forall d \in D, c \in C, w \in W \\
Z_{dcw} &\leq Y_{cw}, \quad \forall d \in D, c \in C, w \in W \\
Z_{dcw} &\geq X_{dc} + Y_{cw} - 1, \quad \forall d \in D, c \in C, w \in W.
\end{aligned} \tag{9}$$

We set binary variables  $T_{cvw}$  equal to one when both ward  $v \in W$  and  $w \in W$  are assigned to cluster  $c \in C$ , and zero otherwise. The correct value of  $T_{cvw}$  is determined by constraints similar to constraints (9).

Except from the quadratic part, the objective function (6) also contains a min-max term. The maximum of  $\sum_{d \in D} X_{dc}$  over  $c$  can easily be determined by adding an extra constraint to the formulation.

Concluding, the described problem can be solved to optimality by solving an integer linear program. However, the solving time is large since based on the above described modeling techniques the resulting instances are already quite large for moderate input instances. Therefore, we discuss several fast heuristic solution methods in the next subsection.

## Heuristic Solution Methods

To be able to solve the considered problem in reasonable computation time, we consider two heuristic solution methods. The first method approximates the stepwise function  $f_B(\lambda\mu)$  which might reduce the computation time as the number of integer variables reduces. The second method solves the problem in two phases. The first phase clusters the clinical departments and the second phase assigns the clusters to wards. By splitting up the problem, we hope to reduce the computation time.

### Approximation Solution Method

Figure 2 already indicates that the stepwise function  $f_B(\lambda\mu)$  is ‘close’ to linear in the considered interval except at the origin. This indicates that it might be possible to approximate the stepwise function by a linear function to reduce the computation time. To further investigate the linear behavior of the stepwise function, we consider the step sizes, i.e., the lengths of the intervals for which the number of beds needed is constant, for both evaluated values of  $B$ , which are shown in Figure 3.

As the slope of the stepwise function is higher for  $B = 0.05$  than for  $B = 0.1$ , it is no surprise that the step sizes are higher for  $B = 0.1$  when compared to  $B = 0.05$ . In addition, Figure 3 shows that the step sizes converge rather quickly for both values of  $B$ . When the number of beds is 20 or more, the step size is almost constant which indicates an almost constant slope. As most wards consist of more than 20 beds, we may easily approximate the stepwise function by a linear function.

Shakhov (Shakhov, 2010) proved that the number of beds  $x$  needed can be approximated by  $\lambda\mu(1 - B)$ . However, this approximation only holds when  $\lambda\mu$  is sufficiently large. As we only want to approximate  $x$  for the interval considered in Figure 2, we introduce the following linear approximation.

Because we want to guarantee the blocking probability  $B$ , we do not want to underestimate the number of beds needed. Therefore, we determine a linear function which never underestimates the stepwise function in the considered interval when rounded up to the nearest integer. In addition, we minimize the maximum overestimation of the stepwise function in the considered interval, because this improves the quality of the solution when compared to the optimal solution. When the overestimation is too large, it may happen that a certain cluster cannot be assigned to a ward, because the number of available beds seems to be too small. However, when the number of needed beds is determined exactly, i.e. by the stepwise function, the number of



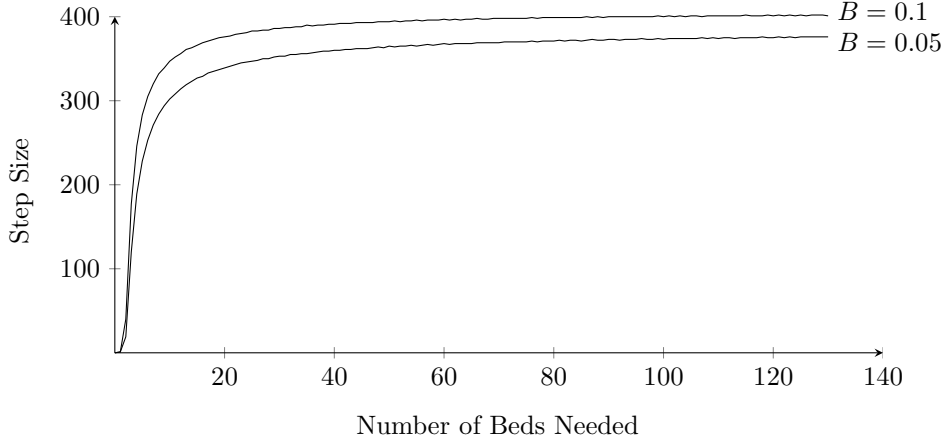


Figure 3: Step sizes of stepwise function

available beds might be sufficient. Therefore, we minimize the maximum overestimation to approximate the optimal solution as good as possible.

We use the following mixed integer linear program (MIP) to determine the linear function which satisfies the two stated criteria, where  $P$  and  $R$  are the coefficients of the linear function used to approximate the stepwise function. Note, that to linearize the given constraint and objective function, we need additional integer variables.

$$\begin{aligned}
 \min \quad & \max_{(\lambda\mu)} [P(\lambda\mu) + R] - f_B(\lambda\mu) \\
 \text{s.t.} \quad & [P(\lambda\mu) + R] \geq f_B(\lambda\mu), \forall \lambda\mu \\
 & P, R \in \mathbb{R}
 \end{aligned} \tag{10}$$

Figure 4 shows the stepwise function and linear approximation for  $B = 0.05$ . For the considered interval, the maximum overestimation equals 4 beds, which is achieved at the beginning and end of the interval, i.e., when less than 0.05 or more than 120 patients are admitted per day. Note, that in practice the number of beds needed by a cluster often lies between 20 and 60. As Figure 4 shows, the overestimation is much smaller in this region, i.e., at most one bed.

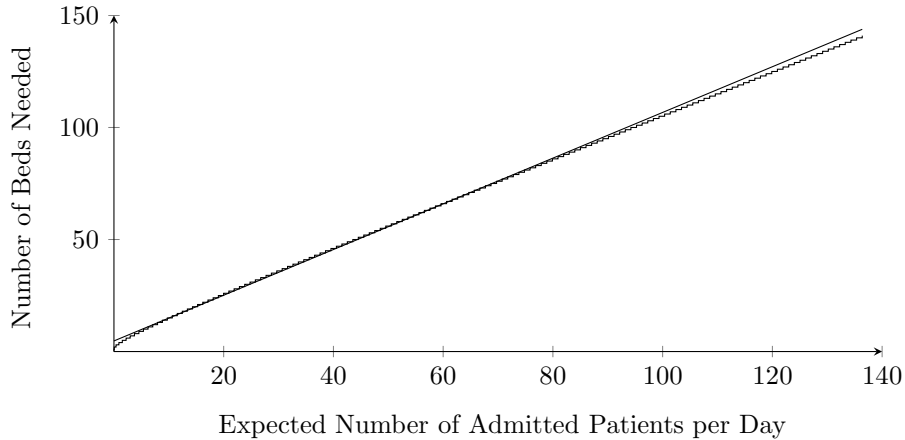


Figure 4: Stepwise function and approximation stepwise function for  $B = 0.05$

Using the above linear approximation, we simply add the following constraint to the formulation given in

Section to determine the number of required beds:

$$V_c = P(\lambda\mu)_c + R, \forall c \in C. \quad (11)$$

Note, that we do not need to round the number up as the number of beds available on the assigned wards should be larger than or equal to the number of beds needed by the cluster.

By using this linear approximation function, we can eliminate the binary variables and the two constraints needed to determine the right interval of the stepwise function. However, to ensure that  $V_c = 0$  when cluster  $c \in C$  is empty, we do have to add binary variables which denote whether cluster  $c \in C$  is empty or not. To determine the correct value of these binary variables, we need two constraints for each cluster  $c \in C$ . This means that the number of constraints stays the same, but instead of a binary variable for each cluster and interval pair, we only need a binary variable for each cluster. Thus, the number of binary variables reduces significantly when an approximation of the stepwise function is used.

### Hybrid Heuristic

As already mentioned in Section , the considered problem can be divided into two phases. In the first phase, clinical departments are clustered and in the second phase, the clusters are assigned to wards. This approach has some similarities with column generation (Dantzig and Wolfe, 1960), which is an iterative process of generating columns for a linear program and selecting a subset of these columns. However, in our case, we do not only need to selected a subset of clusters such that every clinical department is chosen once, but we also need to assign the clusters to one or more wards to guarantee a feasible solution. Therefore, the column generation approach in its original form is not suitable for the problem considered in this paper. However, the basic idea of solving the problem in two phases is an interesting approach.

In the column generation approach, new columns are generated on the fly. Applying this to our problem would mean that each step in the iterative process consists of two phases, where in the first phase new clusters are added to the already generated set of clusters, and in the second phase, the new complete set of clusters is used to select a subset of these clusters. We modify this approach as we completely separate the first and second phase, i.e., we do not perform both phases in an iterative manner. We first execute the first phase by generating a large set of feasible clusters. In the second phase, we select a subset of these clusters which forms a partition of the clinical departments and assign them to wards. As the set of clusters generated in the first phase is too large to completely be considered in an ILP approach for the second phase, we perform a local search method which only considers a subset of this large set in each step. Note, that during the local search procedure, the large set is not changed. Only the subset considered in each step of the local search procedure changes.

The first phase in the two phase approach is generating clusters or in other words assigning clinical departments to clusters. The only constraints considered when forming the clusters in the first phase are the medical limitations and that there must be at least one subset of clusters that forms a partition of the clinical departments. When the complete generation process is done randomly, there is a risk that no subset of clusters exists which form a partition of the clinical departments, and thus, no feasible solution will exist. Therefore, we first generate two sets of clusters which guarantee that such a partition exists. The first set of clusters represents a partition of the clinical departments for which a small number of beds is needed. This partition should ensure that with high probability a feasible assignment of clusters to wards exists. Note, that in general a small number of clusters leads also to a small number of beds needed. Therefore, we search for a clustering with a minimal number of clusters. This set of clusters can easily be generated by solving the following ILP for

$|C| = 1$ ,  $|C| = 2$ , etc. until a feasible solution is found.

$$\begin{aligned}
\sum_{c \in C} X_{dc} &= 1, \quad \forall d \in D \\
X_{dc} + X_{ec} - 1 &\leq a_{de}, \quad \forall c \in C \text{ and } \forall d, e \in D, \\
X_{dc} &\in \{0, 1\}, \quad \forall d \in D \text{ and } \forall c \in C.
\end{aligned} \tag{12}$$

We also introduce set of clusters, where each clinical department forms a cluster on its own. These clusters might be useful in finding feasible partitions of the clinical departments when the clusters are assigned to the wards.

As the clusters described above most likely will not lead to an optimal solution, we also generate random clusters. For each of these  $N$  clusters, we start with selecting one of the clinical departments at random. Next, we determine the set  $\bar{D}$  of clinical departments that are allowed to share a ward with the selected clinical department and thus, are allowed to be assigned to the same cluster. To determine the (maximum) size of the generated cluster, i.e., the number of clinical departments allowed to be assigned to this cluster, we randomly determine a number between one and the cardinality of  $\bar{D}$ . Then, we randomly add clinical departments from  $\bar{D}$  to the cluster until the maximum size is achieved or until none of the not yet assigned clinical departments from  $\bar{D}$  can be added due to the medical limitations. When a generated cluster is the same as one of the previously generated clusters, we start the process again to ensure that all  $N$  generated clusters are unique. The process of generating the  $N$  random clusters is explained into more detail by the following algorithm.

Repeat until  $N$  clusters are added to set  $C$ :

1. Open a new cluster  $\bar{c}$ .
2. Select a clinical department  $d \in D$  at random and assign it to cluster  $\bar{c} \in C$ , i.e.,  $X_{d\bar{c}} = 1$ .
3. Determine set  $\bar{D} \subseteq D$  of clinical departments which are allowed to be assigned to the same cluster as the selected clinical department  $d$ . Note, that  $d \notin \bar{D}$ .
4. Randomly draw a number from  $\{2, \dots, |\bar{D}|\}$  to determine the (maximum) size  $S$  of cluster  $\bar{c} \in C$ . Note, that we do not generate clusters of maximum size or of size one as these these clusters are already present in the two extreme solutions.
5. Repeat until  $\sum_d X_{d\bar{c}} = S$  or  $\bar{D} = \emptyset$ :
  - (a) Select a clinical department  $d \in \bar{D}$  at random and assign it to cluster  $\bar{c} \in C$ , i.e.,  $X_{d\bar{c}} = 1$ .
  - (b) Reduce  $\bar{D}$  to the set of clinical departments which are allowed to be assigned together with all clinical departments in  $\bar{c}$ .
6. If  $\bar{c} \notin C$ , add  $\bar{c}$  to  $C$ , i.e.,  $C := C \cup \{\bar{c}\}$ .

After this phase 1, we need to determine which combination of clusters leads to the best solution. As the chosen number  $N$  of generated clusters is generally quite large to guarantee a good feasible solution, selecting the best clusters from this set with the use of an ILP would take too long. Therefore, we have developed a local search procedure which solves a smaller ILP in each iteration. More precisely, in each iteration of the local search approach, we select a subset of the generated clusters  $\bar{C} \subset C$  and use an ILP to determine which set of the selected clusters leads to the best solution when considering only the clusters in  $\bar{C}$ . The ILP used for this

step is a reduced version of the ILP formulation introduced in Subsection . To determine whether a cluster is used in the optimal solution of this ILP, we need extra binary variables  $\gamma_c$  which are one when cluster  $c \in \bar{C}$  is selected, and zero otherwise. Note, that as the clusters are generated before solving this ILP, variables  $X_{dc}$  are known and fixed, and the number of beds  $V_c$  needed by cluster  $c \in \bar{C}$  can be determined beforehand. This means that the only free variables are the binary variables  $Y_{cw}$  and  $\gamma_c$ .

$$\min \left( \max_{c \in \bar{C}} \sum_{d \in D} X_{dc} \gamma_c + \sum_{c \in \bar{C}} \sum_{w \in W} \sum_{v \in W} g_{wv} Y_{cw} Y_{cv} - \sum_{c \in \bar{C}} \sum_{d \in D} \sum_{w \in W} h_{dw} X_{dc} Y_{cw} \right) \quad (13)$$

$$\text{s.t.} \quad \sum_{c \in \bar{C}} X_{dc} \gamma_c = 1, \quad \forall d \in D, \quad (14)$$

$$Y_{cw} \leq \gamma_c, \quad \forall c \in \bar{C}, \quad \forall w \in W, \quad (15)$$

$$\sum_{c \in \bar{C}} Y_{cw} \leq 1, \quad \forall w \in W, \quad (16)$$

$$V_c \gamma_c \leq \sum_w Y_{cw} b_w, \quad \forall c \in \bar{C}. \quad (17)$$

The objective function of the ILP is the same as (6) except that it is restricted to the selected clusters instead of all clusters in the chosen subset  $\bar{C}$ . Constraint (14) is introduced to select a set of clusters such that each clinical department  $d \in D$  is selected exactly once. Constraint (15) ensures that only the selected clusters are assigned to one or more wards. Constraint (16) is exactly the same as constraint (4) and constraint (17) differs slightly from constraint (5) to only consider the beds needed by the selected clusters.

To find a good solution to the original problem, we use a local search approach. A local search approach starts with an initial solution and then iteratively moves to a neighbor solution. Hereby, a solution is given by a set  $\bar{C} \subset C$  of  $Q$  clusters and the objective value of this solution is given by the optimal solution of the ILP given by (13)-(17). As starting solution, we select a subset of clusters  $\bar{C}$  which includes the set of clusters for which the clinical departments are partitioned in a minimal number of clusters and the set of clusters where each clinical department has its own cluster. The subset of clusters  $\bar{C}$  for the initial solution is further completed with randomly chosen clusters from  $C$ . For this initial subset of clusters, the ILP given by (13)-(17) is solved. Note, that the described initial solution does not always guarantee that a feasible solution to the ILP exists. However, including the two mentioned sets of clusters increases the probability of a feasible solution existing. When no feasible solution exists, the set of randomly selected clusters is replaced by a new set of randomly selected clusters until a feasible initial solution is found. The neighborhood of a solution  $\bar{C}$  is defined by all solutions  $\tilde{C} \subset C$  which contain the clusters selected in the optimal solution of the ILP belonging to  $\bar{C}$ , i.e., clusters  $c \in \bar{C}$  for which  $\gamma_c = 1$ . By randomly adding clusters from  $C$  to the subset of clusters forming the optimal solution of  $\bar{C}$ , we randomly choose one of the neighbor solutions. The following algorithm further describes the local search approach, where  $Q$  denotes the cardinality of the chosen set  $\bar{C}$ , which is the same for all iterations.

1. Select a subset of the generated clusters  $\bar{C} \subset C$  cardinality  $Q$  that includes the two starting solutions mentioned before.
2. Solve the ILP given by (13)-(17) and let  $\bar{L}$  denote the objective function value of the corresponding optimal solution.

3. Repeat until  $K$  consecutive iterations are executed without reducing  $\bar{L}$ :

- (a) Generate a new subset of clusters  $\bar{C} \subset C$  with  $|\bar{C}| = Q$  that includes the clusters selected in the optimal solution of the ILP in the previous iteration.
- (b) Solve the ILP given by (13)-(17) and let  $L$  denote the objective function value of the optimal solution.
- (c) If  $L < \bar{L}$  set  $\bar{L} := L$ .

Note, since the clusters selected in the previous optimal solution are again in  $\bar{C}$ , we always have a feasible solution for the ILP and the objective function value  $L$  in step 3.(b) is less than or equal to the objective function value of the previous iteration. Therefore, the sequence of the  $\bar{L}$ -values is non-increasing.

## Computational Results

This section aims to compare the three described solution methods with respect to the computation time and the quality of the solutions based on data provided by HagaZiekenhuis, who initiated this research. In addition, we want to show the usefulness and applicability of the methods when used in practice.

The used data consists of 16 clinical departments and 13 wards for which the number of beds varies from 8 to 32. The lay-out of the hospital is depicted in Figure 5, where the elevator and stairs are positioned in the middle of side A and B. The number of beds available on each ward is depicted between brackets. Note, that 16 wards are shown, however, wards 9A, 8A and 8B are not equipped to admit patients. When two wards are on the same floor, the distance between these two wards is zero. When two wards are on a different floor but on the same side of the building, i.e., both are on side A or B, the distance is the distance in floors. When the two wards are on opposite sides of the building and on different floors, the distance equals the distance in floors plus one.

11A (28)	11B (32)
10A (32)	10B (32)
<del>9A</del>	9B (32)
<del>8A</del>	<del>8B</del>
7A (32)	7B (32)
6A (32)	6B (32)
5A (32)	5B (30)
4A (8)	4B (24)

Figure 5: Lay-out wards and number of beds available

The terms of the objective function might be conflicting, e.g., more clinical departments in one cluster may reduce the sum of the distances between clusters. However, in practice, we want to make a fair trade-off between these conflicting objectives. Therefore, we multiply the first term by 10 such that this part of the objective has the same value range as the other two terms.

The medical restrictions are that all surgical departments can be clustered and all non-surgical departments can be clustered. Furthermore, the cardiology department cannot be in the same cluster as any of the other departments as the cardiology department is quite large. Therefore, the number of clusters in the optimal solution for our problem instance is at least three. When the blocking probability is set to 0.95, the total number of beds needed for this solution with only three clusters is 321, which therefore is a lowerbound on the

number of beds needed. When the clinical departments are not clustered at all, we obtain an upperbound on the number of beds needed which equals 365. The total number of beds available on the wards equals 378. This means that in theory, the number of beds is sufficient to accommodate all patients when none of the clinical departments are clustered. However, this does not guarantee that we have a feasible solution as we still have to find a feasible clustering and assignment of clusters to wards.

During testing, we realized that the given ILP formulations lead to a large number of symmetric solutions and that this leads to large computation times for solving the ILP. Therefore, we first discuss the use of symmetry-breaking constraints in Section . In Section , we report on experiments with different values for the parameters used in the hybrid heuristic to determine which values result in the best performance. In Section , we discuss the computational results of the considered solution methods with varying blocking probabilities. In Section , we investigate the effect of a change in the uncertain parameters  $\mu$  and  $\lambda$  by considering several scenarios.

All three solution methods are implemented in AIMMS 3.11 and run on an Intel Core2 Duo CPU P8600 2.40 GHz with 3.45 GB RAM. The ILPs are solved with CPLEX 12.3.

## Symmetry-Breaking Constraints

The ILP formulated in Section is completely symmetric with respect to the clusters. For any solution, an equivalent solution can be constructed by switching the clinical departments assigned to any pair of clusters. This results in  $|C|!$  equivalent solutions based on the permutation of clusters. The number of symmetric solutions can be reduced if we order the clusters with respect to one of its characteristics. We consider the following three characteristics: (1) the number of clinical departments assigned to a cluster, (2) the value of  $(\lambda\mu)_c$  which represents the expected number of patients treated on a given day, and (3) the number of beds needed for a cluster to guarantee the predefined blocking probability. The ordering with respect to these three characteristics can be formulated by the following constraints:

$$\sum_{d \in D} X_{dc} \geq \sum_{d \in D} X_{d,c-1}, \forall c \in C \text{ and } c > 1 \quad (18)$$

$$\sum_{d \in D} X_{dc} \lambda_d \mu_d \geq \sum_{d \in D} X_{d,c-1} \lambda_d \mu_d, \forall c \in C \text{ and } c > 1 \quad (19)$$

$$V_c \geq V_{c-1}, \forall c \in C \text{ and } c > 1 \quad (20)$$

Constraint (18) represents the ordering according to the number of clinical departments assigned to a cluster, constraint (19) the ordering according to the expected number of patients treated per day, and constraint (20) the ordering according to the number of beds needed for each cluster. We tested these three constraints for both the exact formulation and the approximation solution method. The results of these tests are shown in Table 1 which shows the runtime in seconds, the obtained objective function value, and, in case the optimal solution is not found within 10 hours, the LP-bound.

The symmetry-breaking constraints do not have the same effect on both solution methods, because of the differences in the ILP formulations. The symmetry-breaking constraint that reduces the computation time the most for the exact formulation is the constraint that sorts the clusters according to the expected number of patients treated per day, i.e., constraint (19). The symmetry-breaking constraint that reduces the computation time the most for the approximation solution method is the constraint that sorts the clusters according to the number of beds needed, i.e., constraint(20). Therefore, we add these constraints to the ILP formulation of the exact and approximation solution method.

<b>Exact Method</b>	<b>Runtime</b>	<b>Objective</b>	<b>LP-bound</b>
# Clinical departments	36000	–	20
# Patients treated per day	1899	57	57
# Beds	6450	57	57
<b>Approximation Method</b>	<b>Runtime</b>	<b>Objective</b>	<b>Lowerbound</b>
# Clinical departments	3290	57	57
# Patients treated per day	5609	57	57
# Beds	866	57	57

Table 1: Computational results symmetry-breaking constraints

## Parameter Setting Hybrid Heuristic

For the hybrid heuristic, we have to determine several parameter values. The first parameter is  $N$ , the number of randomly generated clusters. As for the considered instance the number of feasible clusters is quite small, namely 575, we choose to consider all feasible clusters. As mentioned before, the minimum number of clusters needed for the instance is three. The first group consist of only the cardiology department and therefore, there is only one feasible cluster for this group. The second group is the group of non-surgical departments. This group consists of six different departments and thus, we can generate  $2^6$  different clusters for this group. The third group consist of the nine surgical departments which results in  $2^9$  extra clusters. In total this results in  $1 + 2^6 + 2^9 = 575$  different feasible clusters.

Then, there are two remaining parameter values that need to be determined. The first is the number of clusters  $Q$  to select in each step, and the second is the stopping criterion  $K$ . To determine the best value for these parameters, we performed several tests by solving the problem 25 times for several parameter settings. For each parameter setting, we have determined the number of times the optimal solution is found, the standard deviation of the objective values and the minimum, average and maximum runtime in seconds. The computational results are given in Table 2.

<b>Q</b>	<b>K</b>	<b># Optimal</b>	<b>Std. Dev. Objective</b>	<b>Minimum Runtime (s)</b>	<b>Average Runtime (s)</b>	<b>Maximum Runtime (s)</b>
40	25	13	6.4	149	291	409
40	50	15	8.6	354	519	772
40	75	21	8.7	430	764	1409
40	100	19	2.6	572	828	1223
50	25	17	6.5	241	372	586
50	50	20	2.4	434	706	1303
50	75	23	1.6	690	1080	1870
50	100	23	1.6	775	1102	2040
60	25	19	6.5	321	507	813
60	50	22	1.9	631	1015	2674
60	75	25	0	948	1337	2047

Table 2: Parameter settings hybrid heuristic

As expected, the runtime increase when  $Q$  or  $K$  increases as more iterations are performed. In addition, the number of times the optimal solution is found also increases with  $Q$  and  $K$ . Therefore, we should make a trade-off between the runtime and the quality of the solution found. We choose to set  $Q$  to 60 and  $K$  to 75 as

these settings provides the best quality of the solution found. In addition, the runtime for these settings is still reasonable. For the further computational tests, these values are chosen.

## Comparing Solution Methods

To compare the three proposed solution methods, we have used the methods to solve the instance described in the introduction of Section with blocking probability 0.05 and 0.1. In this way, we also get a feeling about the influence of the blocking probability on the obtained results. Table 3 shows the results for the three proposed solution methods for blocking probability 0.05 and 0.1. For each method and blocking probability combination, we show (1) the total computation time in seconds, (2) the objective function value, (3) the number of clusters used, (4) the maximum number of clinical departments clustered into one cluster, (5) the total distance between assigned wards, (6) the total fulfilment of preferences, (7) the number of beds needed according to the Erlang loss model, and (8) the number of beds needed according to the approximation.

$B = 0.05$	Exact Method	Approximation Method	Hybrid Heuristic
Time (s)	1899	866	1175
Objective	57	57	57
# Clusters	5	5	5
$\max_c \sum_d X_{dc}$	5	5	5
$\sum_{c,v,w} g_{vw} Y_{cw} Y_{cv}$	22	22	22
$\sum_{d,c,w} h_{dw} X_{dc} Y_{cw}$	15	15	15
# Beds exact	335	336	335
# Beds approx.	–	338	–
$B = 0.1$	Exact Method	Approximation Method	Hybrid Heuristic
Time (s)	> 10 hours	1773	796
Objective	36	36	36
# Clusters	6	6	6
$\max_c \sum_d X_{dc}$	3	3	3
$\sum_{c,v,w} g_{vw} Y_{cw} Y_{cv}$	16	16	16
$\sum_{d,c,w} h_{dw} X_{dc} Y_{cw}$	10	10	10
# Beds exact	309	309	309
# Beds approx.	–	311	–

Table 3: Results all methods with varying blocking probabilities

When the blocking probability increases, less beds are needed by the clusters. Therefore, we expect that increasing the blocking probability will improve the achieved solution with respect to the objective function value. The results depicted in Table 3 confirm this as the objective function value decreases from 57 to 36. This means that instead of clustering five clinical departments only three departments are clustered. This leads to an increase in the number of clusters formed. In addition, the total distance between assigned wards decreases from 22 to 16. The solution obtained when the blocking probability is set to 0.1 performs slightly worse when we look at fulfilling the preferences of a clinical department for a certain ward as this value decreases from 15 to 10.



Table 3 shows that all considered solution methods leads to the same objective function value. Thus, we can conclude that both heuristic solution methods perform well and thus, provide good solutions to the original problem.

Abbr.	Description	Abbr.	Description	Abbr.	Description
CAR	Cardiology	INT	Internal medicine	OTO	Otolaryngology
DER	Dermatology	NEU	Neurology	PLA	Plastic surgery
GAS	Gastroenterology	OPH	Ophthalmology	PUL	Pulmonary medicine
GEN	General surgery	ORA	Orthognathic surgery	RHE	Rheumatology
GER	Geriatrics	ORT	Orthopedics	URO	Urology
GYN	Gynecology				

Table 4: Abbreviations clinical departments

However, there are also some differences between the solutions found by the three solution methods. All solutions have the same objective function value, but the formed clusters and the assignment of clusters to wards differs for each of the three methods. Thus, there are several configurations which lead to the same objective function value. Extra terms might be added to the objective function to make a distinction between these solutions. Figure 6 shows the obtained solutions for all three solution methods. The used abbreviations in Figure 6 are described in Table 4.

Exact Method		Approximation Method		Hybrid Heuristic		
	A	B	A	B	A	B
11	NEU-ORT-OTO PLA	NEU-ORT-OTO PLA	NEU-OPH-ORA PLA	NEU-OPH-ORA PLA	NEU-ORA-OTO PLA-URO	NEU-ORA-OTO PLA-URO
10	GER-INT-RHE	GER-INT-RHE	GEN-GYN-ORT OTO-URO	GEN-GYN-ORT OTO-URO	GEN-GYN-OPH ORT	GEN-GYN-OPH ORT
9		GER-INT-RHE		GEN-GYN-ORT OTO-URO		GEN-GYN-OPH ORT
8						
7	DER-GAS-PUL	DER-GAS-PUL	DER-GAS-GER PUL	DER-GAS-GER PUL	GAS-GER-PUL	GAS-GER-PUL
6	GEN-GYN-OPH ORA-URO	GEN-GYN-OPH ORA-URO	INT-RHE	INT-RHE	DER-INT-RHE	DER-INT-RHE
5	CAR	CAR	CAR	CAR	CAR	CAR
4	GEN-GYN-OPH ORA-URO	CAR	CAR	INT-RHE	CAR	DER-INT-RHE

Figure 6: Lay-out hospital 0.95 blocking probability

We also see differences between the three solution methods with respect to the computation time. The computation time for the exact solution method is the longest for both chosen values of the blocking probability when compared to the two heuristic solution methods. We also see that the computation time for the exact solution method increases significantly when the blocking probability is changed from 0.05 to 0.1. This indicates that even the order of magnitude of the computation time for the exact solution method cannot easily be predicted beforehand which makes this method somehow impractical. The computation times for the two heuristic methods also changes with the blocking probability. When the blocking probability is set to 0.05 the

approximation solution method is faster, but when the blocking probability is set to 0.1 the hybrid heuristic is faster. Because it is not clear from these results which of the two heuristic solution methods performs better overall, we have used both methods to evaluate the scenarios described in the next section.

## Evaluating Scenarios

As the average LOS  $\mu$  and the expected number of admissions per day  $\lambda$  are very uncertain parameters which may change over the years, it is necessary to evaluate several what-if scenarios. These evaluations can be used to find a solution which is robust to foreseen changes in  $\mu$  and  $\lambda$ .

For the considered instance, we only know  $\mu\lambda$ , i.e., the expected number of patients admitted per day, and not the values of  $\mu$  and  $\lambda$  separately. Therefore, we cannot investigate changes in  $\mu$  or  $\lambda$ , however, we can investigate a change in  $\mu\lambda$ . Let us assume that the growth of each clinical department lies between 0.75 and 1.25, i.e.,  $\mu\lambda$  can reduce to at most  $0.75\mu\lambda$  and increase to at most  $1.25\mu\lambda$ . Then, we can determine the effect on the obtained solution when the blocking probability is set to 0.05. When  $\mu\lambda$  reduces to  $0.75\mu\lambda$  for some of the clinical departments and none of the other clinical departments grows, the obtained solution in Section with blocking probability 0.05 is still feasible. Moreover, in many cases a better solution can be found because less beds are needed. Table 5 shows the results for the two heuristic solution methods when for each clinical department  $\mu\lambda$  is reduced to  $0.75\mu\lambda$ .

$B = 0.05$	Approximation Method	Hybrid Heuristic
Time (s)	851	263
Objective	20	20
# Clusters	6	8
$\max_c \sum_d X_{dc}$	3	3
$\sum_{c,v,w} g_{vw} Y_{cw} Y_{cv}$	0	0
$\sum_{d,c,w} h_{dw} X_{dc} Y_{cw}$	10	10
# Beds exact	263	268
# Beds approx.	265	–

Table 5: Results when  $\mu\lambda$  is reduced to  $0.75\mu\lambda$

Table 5 indeed shows that the objective function value is reduced from 57 to 20. This reduction is caused by the fact that the number of clinical departments clustered is reduced from 5 to 3 which in turn leads to an increase in the number of clusters formed. The solution obtained by the approximation solution method uses 6 clusters and the hybrid heuristic uses 8 clusters. We also see that the distance between assigned wards is reduced to 0 for all formed clusters, but that the preferences are fulfilled less as this value is reduced from 15 to 10. For this scenario, we see that the hybrid heuristic takes less time than the approximation solution method.

When some of the clinical departments grows, i.e., the value of  $\mu\lambda$  increases, the solution obtained in Section with  $B = 0.05$  may become infeasible. This is the case when the number of beds needed by a cluster exceeds the number of beds available on the assigned wards. To see when this problem may occur, we present in Table 6 for all three solution methods the clinical departments assigned to a cluster, the number of beds needed by this cluster, and the number of beds assigned to this cluster.

Table 6 shows how many slack is available for each cluster. As slack we define the number of spare beds available at the assigned wards. Some clusters already need all assigned beds, but some have more than 20 extra

beds available. The more spare beds are available, the more a cluster can grow in the coming years. However, when a cluster with none or only a few spare beds grows, the obtained solution might become infeasible. This does not necessarily mean that no feasible solution can be found, but that the generated clusters and the assignment of these clusters to wards may have to be changed. However, when too many clinical departments grow it might be the case that no feasible solution is available. For the considered case, there is no feasible solution when  $\mu\lambda$  increases to  $1.25\mu\lambda$  for all clinical departments. Therefore, in the following, we determined the maximum possible growth for all departments such that there is still a feasible solution available.

	<b>Exact Method</b>	<b>Approximation Method</b>	<b>Hybrid Heuristic</b>
Cluster 1: Departments Needed/Assigned	INT-RHE-GER 80/96	INT-RHE 80/88	INT-RHE-DER 80/88
Cluster 2: Departments Needed/Assigned	GEN-GYN-OPH-ORA-URO 72/72	GEN-GYN-ORT-OTO-URO 78/96	GEN-GYN-OPH-ORT 75/96
Cluster 3: Departments Needed/Assigned	CAR 65/86	CAR 65/70	CAR 65/70
Cluster 4: Departments Needed/Assigned	GAS-PUL-DER 63/64	GAS-PUL-DER-GER 63/64	GAS-PUL-GER 63/64
Cluster 5: Departments Needed/Assigned	NEU-PLA-ORT-OTO 55/60	NEU-PLA-OPH-ORA 50/60	NEU-PLA-ORA-OTO-URO 52/60

Table 6: Solutions provided by solution methods including number of beds needed/available

Note, that the minimum number of clusters needed is three as described in the introduction of this section. The total number of beds needed is minimal when the clinical departments are clustered as much as possible. Thus, to determine the maximal growth of the clinical departments we consider this extreme solution with only three clusters. The first cluster only consists of the cardiology department, the second cluster consist of all remaining surgical departments, i.e., GEN, GYN, NEU, OPH, ORA, ORT, OTO, PLA and URO, and the third cluster consists of all remaining non-surgical departments, i.e., DER, GAS, GER, INT, PUL and RHE. We have considered three scenarios. For each of the three scenarios, we allow maximal growth, i.e.,  $1.25\mu\lambda$ , for two of the clusters and we determine the maximum allowed growth for the third cluster such that there is still a feasible solution. For the two clusters with maximal growth, we can determine the assignment of wards to these clusters such that a feasible solution is obtained with as few beds as possible. This means that the total slack on the assigned wards is minimized. The total number of beds available on the remaining wards is then available for the third cluster, and thus, we can determine the maximum allowed growth for this third cluster. When the maximum growth is determined, we use the two heuristic solution methods to find the optimal solution with the given input parameters  $\mu\lambda$ . The assignment of clusters to wards may be different than in the assignment determined beforehand, because a better solution may exist when the original objective function is considered.

Table 7 shows the results for these three scenarios. All of these three scenarios are infeasible according to the approximation method, because the number of beds needed for each cluster is overestimated. For the approximation method, the row ‘# Beds exact’ denote the number of beds needed when three clusters are used. The row ‘# Beds approx.’ denotes the number of beds needed for the three clusters when the approximation function is used. Because the clusters are relatively large, the number of beds needed is overestimated more than was the case for the original instance. This phenomena can easily be seen in Figure 4. Because of this overestimation for each of the three clusters, the total number of beds needed by the clusters, lies quite close to the number of beds available, i.e., 378, and therefore, no feasible solution can be found. The hybrid heuristic

$B = 0.05$	Approximation Method			Hybrid Heuristic		
	Scenario 1	Scenario 2	Scenario 3	Scenario 1	Scenario 2	Scenario 3
Time (s)	–	–	–	228	209	333
Objective	Infeasible	Infeasible	Infeasible	165	93	112
# Clusters	–	–	–	3	4	4
$\max_c \sum_d X_{dc}$	–	–	–	9	6	6
$\sum_{c,v,w} g_{wv} Y_{cw} Y_{cv}$	–	–	–	90	48	62
$\sum_{d,c,w} h_{dw} X_{dc} Y_{cw}$	–	–	–	15	15	10
# Beds exact	369	363	365	369	370	373
# Beds approx.	378	371	375	–	–	–

Table 7: Results for the three maximum growth scenarios

uses the exact method to determine the number of beds needed. Therefore, this number is not overestimated, and thus, a feasible solution can be found. For two of the scenarios, even a feasible solution with four clusters is found. Because the number of beds needed almost equals the number of beds available, not much flexibility is left to improve the assignment of clusters to wards. Therefore, the total distance between wards assigned to a cluster is quite large. These results show that when the clusters are relatively large the hybrid heuristic is preferred as this method is better in finding a feasible solution. In addition, the computation time is very fast which makes this method applicable in practice, as several scenarios can be evaluated within a short period of time.

## Conclusions and Recommendations

We have introduced a problem, which in this combination has not been considered in literature before. We cluster one or more clinical departments to reduce the number of beds needed and assign these clusters to wards such that the resulting blocking probability is below a given threshold. The clinical departments are assigned to exactly one cluster such that the clusters form a partition of the clinical departments. Only one cluster is allowed to be assigned to a ward such that only clustered clinical departments share beds. We have developed an exact solution method and two heuristic solution methods to solve this problem.

The exact solution method uses the Erlang loss model to determine how many beds are needed to guarantee a prespecified blocking probability. An ILP is used to determine which departments should be clustered and to assign wards to the formed clusters. A drawback of this exact solution method is that the computation time is unpredictable and quite long. Therefore, we have developed two heuristic solution methods that need less computation time, but still provide good solutions to the original problem.

One of the heuristic solution methods approximates the number of beds needed thereby eliminating the evaluation of the complex Erlang loss formulation. Our tests show that this Erlang loss formulation can easily be approximated by a linear function that minimizes the maximum overestimation of the number of beds. Also for this approach an ILP is used to cluster several clinical departments and to assign the formed clusters to wards. As the number of beds needed by a cluster might be overestimated, we cannot guarantee that this method finds the optimal solution.

The other heuristic solution method uses the exact Erlang loss model to determine the number of beds

needed, but uses a local search approach to determine which clusters should be formed. The assignment of clusters to wards is again done with the use of an ILP. In each iteration of the heuristic, a set of clusters is selected such that each clinical department is assigned to exactly one of these clusters. For this set of clusters, the optimal assignment of these clusters to the wards is determined. The number of beds needed for each cluster is determined beforehand with the use of the Erlang loss model. As the selection of the clusters is determined by a heuristic, this method does not guarantee to find the optimal solution.

All three considered solution methods use the Erlang loss formula or an approximation of this formula to determine the number of beds needed for each cluster. The input for these formulas are the average LOS and the expected number of admissions per day. However, these values are quite uncertain and may change over the years, and thereby, the number of beds needed by a cluster might increase or decrease. Therefore, it is important to evaluate several scenarios before fixing a new lay-out for the hospital.

Our computational results show that the exact solution method takes quite some time and therefore, this method is not useful in practice. The computation time for the two other heuristic solution methods is much shorter and are therefore better applicable in practice. However, the approximation solution method overestimates the number of beds needed for some case, and thus, might claim that no feasible solution exists. But when the number of beds needed is determined exactly, it might be the case that there does exist a feasible solution. Concluding, we believe that the hybrid heuristic is the best solution method for solving this problem as several scenarios can be evaluated within a short amount of time. In addition, the hybrid heuristic always finds a feasible solution when a feasible solution exists. The only drawback is that we cannot guarantee that the optimal solution is found.

A disadvantage of using the Erlang loss formula may be that it only uses the expected number of admission per day. One may claim, that is therefore can only be used in theory, because the number of admissions per day varies during the week. This would mean that the number of beds needed in practice is higher than is determined by the Erlang loss formula. However, several methods exist that level the bed occupancy over the week such that peak occupancies can be reduced or even avoided, see e.g. (van Essen et al., 2012). Thus, the number of beds determined by the Erlang loss model should be enough in practice.

As the input parameters  $\mu$  and  $\lambda$  are quite uncertain, further research should focus on finding robust solutions. As a robust solution we define a solution which is still feasible when the input parameters change in a reasonable way. A possible solution would be to look at a worse case scenario, however, this will lead to a suboptimal solution when the input parameters have different values than in this worst case scenario. Thus, further research should focus on finding solutions which can easily be adapted such that they are near optimal for many or all reasonable values of the input parameters.

## Acknowledgements

This research is supported by the Dutch Technology Foundation STW, applied science division of NWO and the Technology Program of the Ministry of Economic Affairs.

## References

Dantzig G B and Wolfe P (1960). Decomposition principle for linear programs. *Operations Research* **8**(1): 101–111.

- de Bruin A, Bekker R, van Zanten L and Koole G (2010). Dimensioning hospital wards using the erlang loss model. *Annals of Operations Research* **178**(1): 23–43.
- Garey M R and Johnson D S (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co Ltd.
- Gorunescu F, McClean S I and Millard P H (2002a). A queueing model for bed-occupancy management and planning of hospitals. *The Journal of the Operational Research Society* **53**(1): 19–24.
- Gorunescu F, McClean S I and Millard P H (2002b). Using a queueing model to help plan bed allocation in a department of geriatric medicine. *Health Care Management Science* **5**(4): 307–312.
- Green L V and Nguyen V (2001). Strategies for cutting hospital beds: the impact on patient service. *Health services research* **36**(2): 421–442.
- Harper P R and Shahani A K (2002). Modelling for the planning and management of bed capacities in hospitals. *The Journal of the Operational Research Society* **53**(1): 11–18.
- Kokangul A (2008). A combination of deterministic and stochastic approaches to optimize bed capacity in a hospital unit. *Computer Methods and Programs in Biomedicine* **90**(1): 56–65.
- Lapierre S D, Goldsman D, Cochran R and DuBow J (1999). Bed allocation techniques based on census data. *Socio-Economic Planning Sciences* **33**(1): 25–38.
- Li X, Beullens P, Jones D and Tamiz M (2009). An integrated queueing and multi-objective bed allocation model with application to a hospital in china. *Journal of the Operational Research Society* **60**(3): 330–338.
- Qiao S and Qiao L (1998). A robust and efficient algorithm for evaluating erlang b formula. Technical report.
- Shakhov V (2010). Simple approximation for erlang b formula. In *2010 IEEE Region 8 International Conference on Computational Technologies in Electrical and Electronics Engineering (SIBIRCON)*. pp. 220–222.
- Tijms H C (2003). *A First Course in Stochastic Models*. John Wiley & Sons.
- Utley M, Gallivan S, Treasure T and Valencia O (2003). Analytical methods for calculating the capacity required to operate an effective booked admissions policy for elective inpatient services. *Health Care Management Science* **6**(2): 97–104.
- van Essen J T, Bosch J M, Hans E W, van Houdenhoven M and Hurink J L (2012). Improve OR-schedule to reduce number of required beds. Beta Working Paper WP-391, Beta Research School for Operations Management and Logistics, Eindhoven.
- Williams H P (1999). *Model Building in Mathematical Programming*. John Wiley & Sons.