Article

Two-Stage Optimization Methods to Solve the DNA-Sample Allocation Problem

Diego Noceda-Davila [†], Silvia Lorenzo-Freire *,[†] and Luisa Carpente [†]

MODES Research Group, Department of Mathematics, Faculty of Computer Science and CITIC Research Centre, University of A Coruña, 15071 A Coruña, Spain

* Correspondence: silvia.lorenzo@udc.es
† These authors contributed equally to this work.

Abstract: This paper deals with new methods capable of solving the optimization problem concerning the allocation of DNA samples in plates in order to carry out the DNA sequencing with the Sanger technique. These methods make it possible to work with independent subproblems of lower complexity, obtaining solutions of good quality while maintaining a competitive time cost. They are compared with the ones introduced in the literature, obtaining interesting results. All the comparisons among the methods in the literature and the laboratory results have been made with real data.

1. Introduction

Currently, digital transformation, which consists of initiatives to explore the benefits of using modern digital technology, has had a substantial impact on many fields. Digital transformation may affect processes and products and may be applied to organizational structures across a company. Different strategies and potential impacts of digital transformation have been analysed in [1]. In [2,3] reviews regarding different approaches used to implement digital transformation and how those approaches affect companies have been made.

In particular, the healthcare and clinical sector have also been greatly affected by digital transformation. For example, Ref. [4] presented a review on how adopting of modern technologies may impact the performances of healthcare related businesses.

The rise of digital transformation and its adoption has created a situation where competition between different laboratories is intense; labs seek ways to offer the best services to their clients, by reducing operating costs, queues and waiting times. For example, an optimization problem for laboratory staff assignment was developed in [5], and an appointment scheduling problem was studied in [6].

This is the context in which Health in Code (Health in Code: https://www.healthincode.com, accessed on 16 November 2022) operates. Health in Code is a company specialized in cardiovascular genetics, by conducting multiple studies for the detection and prevention of genetic diseases and other pathologies. To carry out these experiments, it is necessary to extract the DNA (abbreviation of deoxyribonucleic acid) of the patients from samples of tissue, saliva, blood, etc. Once the DNA sample has been extracted, it will be treated to sequence the fragment of the DNA sample of interest for the corresponding study. The chosen fragment may depend on what future pathologies should be detected in the patient.

DNA sequencing is the process that is used to determine the order of the four nucleotides, namely, Adenine (A), Cytosine (C), Guanine (G) and Thymine (T), that make up the DNA molecule. This procedure was considered in the Sanger method [7]. The

method, which is also referred to as dideoxynucleotide sequencing and chain termination sequencing, consists of the following steps:

- DNA extraction (from samples of tissue, saliva, blood, etc.) and dilution treatment;
- Polymerase chain reaction (PCR): an effective process for replicating segments of DNA;
- PCR product purification: to remove elements that are used in the PCR process to obtain high-quality DNA samples for sequencing;
- DNA sequencing: to sort the DNA fragments by size in a sequencing machine so that the original piece of DNA can be decoded.

Although all these steps are essential in the Sanger method, this paper focuses on the PCR technique (Kary B. Mullis: https://www.mayoclinicproceedings.org/article/S0 025-6196(11)62225-8/fulltext, accessed on 16 November 2022). The polymerase chain reaction is a method for amplifying DNA to generate millions of copies of one or several pieces of DNA. To perform this reaction, the DNA samples accompanied by chemical reagents are deposited into PCR plates. Then, the plates are placed into thermocyclers (A thermocycler, which is also known as a thermal cycler, is a laboratory machine that allows several temperatures to be set in a block of a plate), which heat the wells up to the reaction temperature of the chemicals. It should be noted that, depending on the DNA fragment to be replicated, the samples must be mixed with a specific reagent and processed at a certain temperature. Moreover, the temperature at which the samples will be processed will also play a key role in determining the position of the samples into the plates.

Laboratories usually employ generic management software, known as LIMS (Laboratory Information Management System), that allows large portions of laboratory tasks to be automated, improving their efficiency. However, in such a specific scenario as the one faced by Health in Code, where many different experiments are being performed, commercial software often suffers from performance issues. In addition, processing each plate is costly both in time and money (up to several thousand euros [PCR Prices: https://sct.uab.cat/genomica-bioinformatica/en/content/prices, accessed on 16 November 2022]). Thus, applying the correct techniques when organizing the experiments, namely, allocating the samples in the PCR plates, could result in substantial savings for the company, allowing it to become more competitive and more attractive to the market.

To solve the DNA-sample allocation problem, Ref. [8] proposed methods capable of greatly improving the behaviour of commercial software. As far as we know, no other framework has addressed this topic. Notice that this problem could remind us of the bin packing problems studied in [9,10] and, in particular, the bin packing with fragmentable items introduced by [11]. However, although in the DNA-sample allocation problem the samples must be packed in the plates, the specific constraints such as the different temperature ranges and the presence of control reagents make not possible to apply the different techniques of bin packing to solve these problems.

Thus, to organize adequately DNA samples in the PCR plates, Ref. [8] developed an integer linear programming model. Since the model could only be solved for small instances, they designed a heuristic based on the simulated annealing technique to deal with bigger instances. Due to the success of the heuristic algorithm, it was implemented in the laboratory, becoming part of a custom LIMS software application developed to meet the specific needs of Health in Code, known as nextLIMS (nextLIMS: https://www.nextlims.com/, accessed on 16 November 2022).

However, even though the method in [8] meets the laboratory requirements, there is room for improvement. So, in this paper new methods to achieve this goal are studied. These methods take advantage of the internal structure of the problem to find new ways of breaking it down, minimizing the complexity of the problems to be solved. They are called two-stage methods, since in the first stage the problems will be solved for each temperature separately. The second stage considers the solutions of the subproblems from the first stage to find a global solution to the original problem. In accordance with this philosophy, two approaches have been proposed: the first one is based on solving an integer linear programming model at each stage and the second combines a heuristic algorithm

at the first stage with the integer linear programming model to find the solutions at the second stage. The new methods allow to obtain better results at a lower computational cost. In addition, since in the first stage several subproblems can be solved separately, different multicore implementation techniques can be applied to reduce the computational times. In this framework, it is shown the utility of several multicore techniques. Thus, new use cases could be addressed, such as long term assignments (e.g., weekly or even monthly scheduling).

The remainder of the paper is structured as follows: Section 2 introduces the problem, accompanied by a revision to the methods presented in [8]. Section 3 provides the details of the developed methods, which constitute the main contribution of this work. The achieved results are presented in Section 4, and Section 5 concludes the paper.

2. The Problem

The problem addressed in this article was previously defined and studied in [8]. In summary, the goal is to arrange a set of DNA samples accompanied by a chemical reagent on laboratory plates in the best possible way. Each well of the plate is filled with a mixture of a DNA sample and a chemical reagent and the plate is then placed in the thermocycler, which heats each of the wells so that the chemicals react. Both the chemical reagent and the processing temperature depend on the fragment of the DNA sample to be sequenced. To understand the choice of different temperatures and chemical reagents for the DNA samples, let us suppose that the laboratory is interested in studying the risk of genetic diseases, for instance. Thus, the laboratory must determine, using the DNA samples of the clients, whether the clients will develop a particular disease. However, since the disease will be detected by evaluating a specified region of DNA, the DNA sample will be mixed with a reagent and processed at the necessary temperature for amplifying this region.

The plates used are of standard size, with 96 wells distributed in 8 rows and 12 columns. However, it should be noted that a thermocycler is not able to set a different temperature for each well; instead, its operation is based on splitting the plates into strips, groupings of 8 rows by 2 columns of wells and setting the temperature of each of these strips. This means that the user must avoid placing chemicals that react at different temperatures in the same strip. Another important detail to take into account is that the difference in temperature between two consecutive strips cannot exceed 5 degrees centigrade due to the operation of the thermalcyclers. Notice that, although the strips in a plate must satisfy these temperature conditions, the temperature in the strips is not fixed, but rather is selected by the user.

It is common to perform the same test on multiple DNA samples, where each sample is mixed with the same reagent; such a set of samples is known as a group. Finally, to ensure that each experiment is performed correctly, it is necessary to reserve a well for one control reagent for each group present on a plate.

Since plate processing is costly in both time and money, the objectives originally defined involve minimizing the number of plates needed (first objective) and keeping the groups together, thus minimizing the total number of occupied wells (second objective). Finally, it involves maximizing the plates occupation rate (third objective). Thus, a solution for the problem indicates where the DNA samples and the corresponding reagents should be placed and it is evaluated by means of the objective functions, that is, the number of occupied plates, the number of occupied cells and the occupation percentage of each plate.

Figure 1 shows an example of a plate as used in the laboratory. In addition to the physical division of the plates into rows and columns, the strips defined by the thermal cycler are marked. Each coloured cell represents a well occupied by a DNA sample, and each of the different colours represents a different group. In addition, the control reagent for each group is always placed first (represented by the wells containing circles). Thus, this plate is composed of 7 groups (indicated by the control reagent at the first cell of each group). The first group is represented by the blue colour and is formed by 11 DNA samples mixed with the reagent and another occupied cell with the control reagent. The cells in

the first strip will be processed at 50 degrees centigrade. In the second and third strip the samples of three different groups can be found (represented by the yellow, the orange and the brown colour). All of them will be processed at the same temperature, 52 degrees centigrade, although the reagents of the groups are different. Notice that the difference between consecutive strips is not greater than 5 degrees centigrade. Moreover, the total number of occupied cells is 74 and the occupation percentage of the plate is 77.08%.

Figure 1. Example of a plate as used in the lab.

2.1. Integer Linear Programming Model

To solve this problem, Ref. [8] proposed an Integer Linear Programming (ILP) model capable of tackling the problem in its entirety, dealing with all the constraints on the presence of control reagents and the difference in temperature between strips.

That ILP model was implemented and tested in this paper; however, the original constraints have been adapted to include another interesting feature of the problem that was not previously considered. The model now allows empty strips to be inserted into a plate when the difference between adjacent temperatures is greater than $5\,^{\circ}$C (for instance, if the difference between two adjacent temperatures is $6\,^{\circ}$C, now is possible to add an empty strip between the strips of both temperatures). This can help to improve the solutions provided by the original model. Throughout this work, it will be referred to as ILP1 and it is presented below.

The model parameters consider the distribution of the set of groups, $\{1,\ldots,n\}$, where N_i represents the number of samples that make up the i-th group. In addition, each group is defined by its reaction temperature, T_i. Since different groups may share the same reaction temperature, it will be introduced the subset of distinct temperatures $\{T^1,\ldots,T^m\} \subseteq \{T_1,\ldots,T_n\}$ with index j. It is also important to know the maximum temperature value present in the problem, T_{max}. Finally, the samples must be placed on a set of plates $\{1,\ldots,p\}$, where each plate, indexed by q, has been assigned the weight w_q.

The variables consider each strip $l \in \{1,\ldots,6\cdot p\}$. For each strip, a temperature t_l must be selected, as well as the number of samples in each group, n_{il}. In addition, it is necessary to know whether the control reagent for a group is present in the strip, which is reflected by using the binary variable x_{il}. Finally, it is necessary to know whether each strip l has been assigned the temperature T^j, which also uses a binary variable, y_{jl}. The ILP1 model is as follows:

$$\min z = \sum_{i=1}^{n}\sum_{q=1}^{p} w_q \sum_{l=6\cdot(q-1)+1}^{6\cdot q} (n_{il} + x_{il}) \tag{1}$$

subject to:

$$\sum_{l=1}^{6\cdot p} n_{il} = N_i, \qquad\qquad \forall i \in \{1,\ldots,n\} \tag{2}$$

$$\sum_{i=1}^{n} (n_{il} + x_{il}) \leq 16, \qquad\qquad \forall l \in \{1,\ldots,6\cdot p\} \tag{3}$$

$$\sum_{j=1}^{m} y_{jl} \cdot T^j = t_l, \qquad\qquad \forall l \in \{1, \ldots, 6 \cdot p\} \tag{4}$$

$$\sum_{j=1}^{m} y_{jl} \leq 1, \qquad\qquad \forall l \in \{1, \ldots, 6 \cdot p\} \tag{5}$$

$$y_{jl} \geq \frac{\displaystyle\sum_{i:T_i=T^j} n_{il}}{n \cdot \displaystyle\sum_{i=1}^{n} N_i}, \qquad\qquad \forall l \in \{1, \ldots, 6 \cdot p\},$$

$$\qquad\qquad\qquad\qquad \forall j \in \{1, \ldots, m\} \tag{6}$$

$$n_{il} - x_{il} \geq 0, \qquad\qquad \begin{array}{l} \forall l \in \{1, \ldots, 6 \cdot p\}, \\ \forall i \in \{1, \ldots, n\} \end{array} \tag{7}$$

$$\sum_{l'=6\cdot(q-1)+1}^{6\cdot q} N_i \cdot x_{il'} - n_{il} \geq 0, \qquad \forall i \in \{1, \ldots, n\},$$

$$\qquad\qquad\qquad \begin{array}{l} \forall l \in \{6 \cdot (q-1)+1, \ldots, 6 \cdot q\}, \\ \forall q \in \{1, \ldots, p\} \end{array} \tag{8}$$

$$\sum_{l=6\cdot(q-1)+1}^{6\cdot q} x_{il} \leq 1, \qquad\qquad \forall i \in \{1, \ldots, n\}, \forall q \in \{1, \ldots, p\} \tag{9}$$

$$t_l + t_{l+k} \leq 5k + T_{max}\left(2 - \sum_{j=1}^{m} y_{jl} - \sum_{j=1}^{m} y_{jl+k}\right) \tag{10}$$

$$\forall k \in \{1, \ldots, 5\}, \forall l \in \{6 \cdot (q-1)+1, \ldots, 6 \cdot q - 1\}, \forall q \in \{1 \ldots, p\}$$

$$t_{l+k} - t_l \leq 5k + T_{max}\left(2 - \sum_{j=1}^{m} y_{jl} - \sum_{j=1}^{m} y_{jl+k}\right) \tag{11}$$

$$\forall k \in \{1, \ldots, 5\}, \forall l \in \{6 \cdot (q-1)+1, \ldots, 6 \cdot q - 1\}, \forall q \in \{1 \ldots, p\}$$

$$t_l \geq 0, n_{il} \in \mathbb{N}, x_{ij} \in \{0,1\}, y_{jl} \in \{0,1\}, \quad \forall i \in \{1, \ldots, n\}, \forall j \in \{1, \ldots, m\}$$

$$\forall l \in \{6 \cdot (q-1)+1, \ldots, 6 \cdot q - 1\}$$

$$\forall q \in \{1 \ldots, p\}$$

Note that, except for the modifications mentioned above, the structure of the ILP1 model is identical to the originally proposed model.

The objective function, (1), minimizes the total cost of sample allocation by multiplying the total occupation of each plate by the weight of the plate. By fixing $w_q = q$ for all q, all three objectives can be satisfied; first, the number of plates to be used is minimized, second, the number of occupied wells is minimized; third, the percentage of occupation of the first plates is prioritized over that of the last plates.

Constraints (2) and (3) ensure that all samples are processed and that no strip capacity is exceeded. Constraints (10) and (11) address the constraint of temperature difference between adjacent strips, which must be less than or equal to $5\,°C$. Notice that these two constraints have been modified from the original model to include intermediate empty strips between samples with widely different temperatures, avoiding the risk of leaving unnecessary gaps in a plate. Constraints (4) and (5) ensure that a valid temperature is assigned to each non-empty strip. Constraint (6) ensures that the temperature of each strip coincides with the reaction temperature of the groups contained within it. Finally,

constraints (7)–(9) guarantee that each group present on a plate is accompanied by exactly one control reagent.

2.2. Heuristic

To solve the performance problems incurred by ILP1, Ref. [8] introduced a heuristic algorithm capable of generating quality solutions at low cost, referred to as Algorithm 1 (A1). This algorithm is based on the simulated annealing technique introduced by [12]. It starts from an initial solution and explores the solution space by executing two different movements.

These movements can be seen in Figures 2 and 3. Figure 2 shows the strip exchange movement, which consists of exchanging two strips whenever the constraints are satisfied. In this example, the third strip of plate 1 is exchanged by the fifth strip of plate 2. This movement has incremented the occupation percentage of plate 1 in detriment of the occupation percentage of plate 2, which could derive in a better solution according to the third objective. Figure 3 shows the grouping movement, which joins samples of the same group, scattered over several plates into one plate. In this case, all the samples of the green group have been moved to plate 1, whereas the samples of the orange group have been moved to plate 2. This movement improves the second objective, since the green group has one control reagent instead of two. Note that any group needs a control reagent in each plate with some sample of this group.
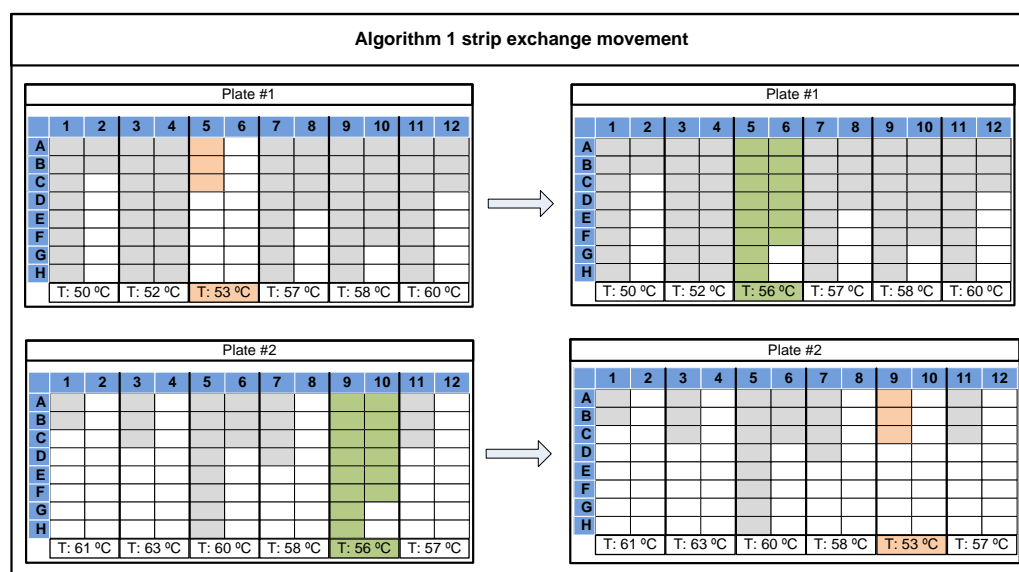

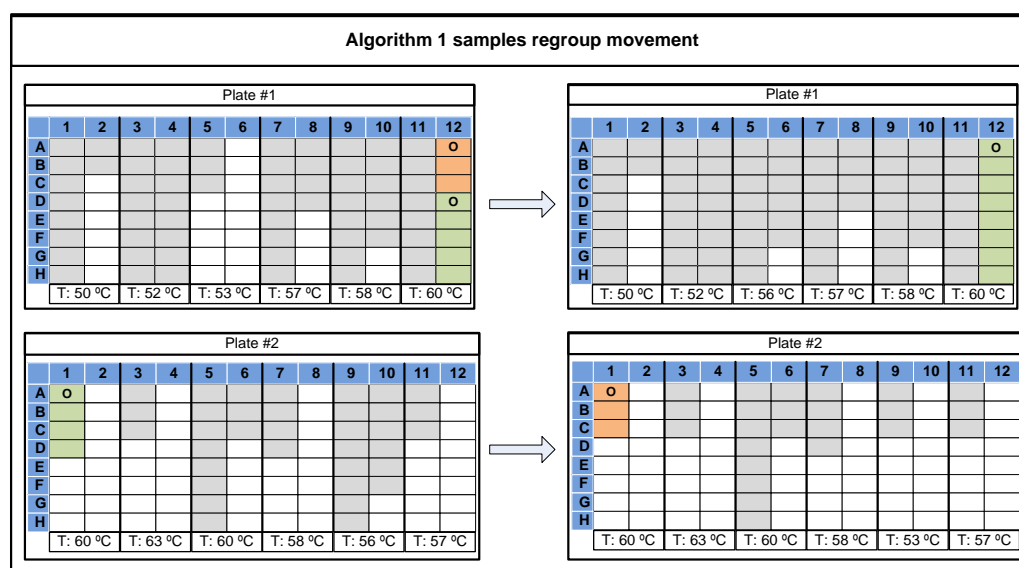**Figure 2.** Algorithm 1 strip exchange movement example.

Figure 3. Algorithm 1 regrouped samples movement example.

3. The Two-Stage Approach

Due to the way the thermocycler operates, samples that require different temperatures are never located in the same strip. This means that groups of samples of different temperatures can be arranged independently. Therefore, the problem can be solved for each temperature separately. The advantage of this approach is that it greatly reduces the complexity of the optimization problems to be solved for two reasons: first, fewer samples must be allocated in each case and second, when all the samples share the same temperature, the constraints that handle the temperature difference between strips do not need to be considered, resulting in simpler models.

Because a global solution is required, all these partial solutions must be merged into a single one that is applicable for the entire instance, by minimizing the number of plates, the total occupation and the occupation of the first plates over that of the last ones.

According to this approach, the problem is solved by using a customized two-stage method. The first stage filters and solves the sample allocation problem for each temperature, which consists of a variant of the bin packing problem with item fragmentation, where each group forms an item. The second stage combines subproblem solutions from the first stage to generate a global solution to the problem that optimizes the considered objectives.

It should be noted that this two-stage method considers all the constraints of the original problem but does not tackle them all at once. Instead, it uses a more intelligent approach to deal with them separately. The first stage focuses on the constraints concerning the control reagent that must accompany the groups on their respective plates and ignores the constraints concerning temperature, which is possible since each subproblem is composed of samples that share the same temperature. Only in the second stage do the temperature constraints come into play. This method can be seen as a two-level variant of the overall allocation problem posed in Section 2.1, working first on one part of the problem (the allocation of the samples into strips) and then on another part (the final location of the strips on the plates).

Figure 4 shows the general structure of the method. The main idea is to filter the problem samples based on the temperatures present in the instance. Each subproblem associated with each temperature is solved separately, that is, the samples that share temperature in a set of plates are assigned, these are accumulated in a global solution and the ordering process is carried out on it to maximize the occupation of the first plates over that of the last plates.
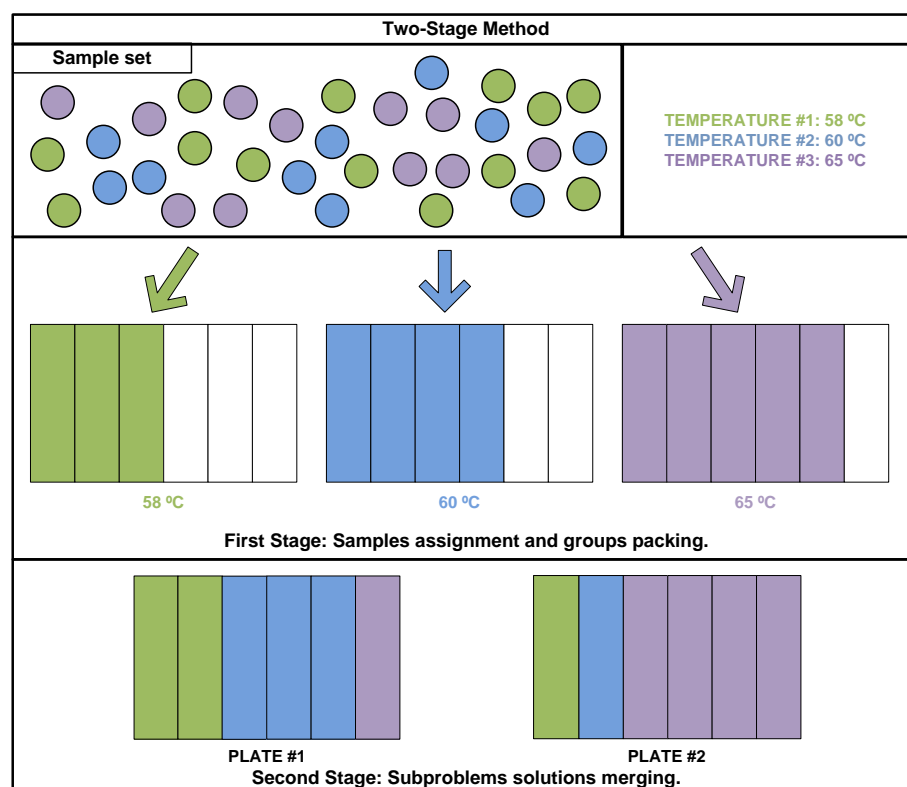
Figure 4. Two-stage method diagram.

The general structure is simple enough to be understood at a glance; the complexity lies in the choice of the best method to solve each stage. Below, the two-stage method is described into detail, as well as the different approximations used to solve it. Figure 5 shows the decomposition of the different methods.

Figure 5. Two-stage method decomposition.

3.1. Method 1. ILP Approach

To reach the optimal solution in each of the stages, a method was developed, henceforth called Method 1 (M1), that uses ILP models to solve each stage, as presented below. The first ILP model solves the problems for the groups of each temperature independently. The second one considers the solutions of the problems at the first stage to find a global solution by rearranging the strips to improve the first and third objectives.

3.1.1. Method 1. First Stage ILP Model

The goal of the first stage is to address groups that share temperature. At this stage, the groups will be fragmented into as few strips as possible. Although this consideration does not seem important when all the samples have to be processed at the same temperature, it plays a fundamental role in the second stage of the method by avoiding unnecessary calculations.

For each temperature T^j, where $j \in \{1, \ldots, m\}$, an ILP model was solved, as shown below. From now on, it will be referred to as ILP2.

Because the temperature T^j is the same for every group in the subproblem, the parameters for the model are the set of groups $\{1, \ldots, n^j\}$, the number of samples N_i that compose them and the set of plates $\{1, \ldots, p\}$ available, in the same way as in ILP1.

As with ILP1, for each strip l, the model variables involve the number of samples assigned from each group, n_{il}^j, as well as the presence of the control reagent, x_{il}. In addition, the presence of the group control reagent at the plate level will also be determined by means of the binary variable z_{iq}. Finally, it is necessary to know if there is at least one sample of a group in the strip, represented by the binary variable v_{il}. It should be noted that no decision needs to be determined regarding temperatures, as those are set by the subproblem itself.

$$\text{Min} \sum_{q=1}^{p} w_q \sum_{l=6(q-1)+1}^{6q} \sum_{i=1}^{n^j} (n_{il}^j + x_{il} + v_{il}) \tag{12}$$

subject to

$$\sum_{l=1}^{6 \cdot p} n_{il}^j = N_i, \qquad\qquad \forall i \in \{1, ..., n^j\} \tag{13}$$

$$\sum_{i=1}^{n^j} (n_{il}^j + x_{il}) \leq 16, \qquad\qquad \forall l \in \{1, \ldots, 6 \cdot p\} \tag{14}$$

$$x_{il} - n_{il}^j \leq 0, \qquad\qquad \forall i \in \{1, \ldots, n^j\}, \forall l \in \{1, \ldots, 6 \cdot p\} \tag{15}$$

$$v_{il} - n_{il}^j \leq 0, \qquad\qquad \forall i \in \{1, \ldots, n^j\}, \forall l \in \{1, \ldots, 6 \cdot p\} \tag{16}$$

$$N_i v_{il} - n_{il}^j \geq 0, \qquad\qquad \forall i \in \{1, \ldots, n^j\}, \forall l \in \{1, \ldots, 6 \cdot p\} \tag{17}$$

$$z_{iq} \leq \sum_{l=6(q-1)+1}^{6q} n_{il}^j, \qquad\qquad \forall i \in \{1, \ldots, n^j\}, \forall q \in \{1, \ldots, p\} \tag{18}$$

$$N_i \cdot z_{iq} \geq \sum_{l=6(q-1)+1}^{6q} n_{il}^j, \qquad\qquad \forall i \in \{1, \ldots, n^j\}, \forall q \in \{1, \ldots, p\} \tag{19}$$

$$\sum_{l=6(q-1)+1}^{6q} x_{il} = z_{iq}, \qquad\qquad \forall i \in \{1, \ldots, n^j\}, \forall q \in \{1, \ldots, p\} \tag{20}$$

$$n_{il}^j \in \mathbb{N} \qquad\qquad \forall i \in \{1, \ldots, n\}, \forall l \in \{1, \ldots, 6 \cdot p\}$$

$$x_{il}, z_{iq}, v_{il} \in \{0, 1\} \qquad\qquad \forall i \in \{1, \ldots, n\}, \forall l \in \{1, \ldots, 6 \cdot p\}$$

$$\forall q \in \{1, \ldots, p\}$$

The objective function, (12), is similar to the objective function of the original model. The only difference is that in this case the group fragmentation among the different strips should be minimized.

Note how most of the constraints are similar to those of the original model, which is the case for constraints (13)–(15) and (18). On the other hand, constraints (16) and (17) are specific to this model and establish the presence of a group in a particular strip, reducing fragmentation. Constraint (19) prevents samples from being placed on a plate without a control reagent. Finally, constraint (20) assures that only one control reagent will be assigned to each plate per group.

This model guarantees that the objectives defined for the original problem will be optimized but it is applied only to the subproblems of the first stage. That is, the total number of non-empty plates is minimized, as well as the total occupation while maximizing

the percentages of occupation. It is important to emphasize that achieving optimality of the subproblems of the first stage does not guarantee optimality for all the objectives in the global model.

All the subproblems generated in this stage are independent and can be solved separately. However, because of the number of samples and groups present, their complexity varies considerably. Although it is possible to find the optimal solution in a short time for most subproblems, for others, finding the optimal solution is computationally expensive.

3.1.2. Method 1. Second Stage ILP Model

In this second stage, a global solution is pursued based on the merged solutions of the subproblems at the first stage. In this step it is necessary to consider the temperature difference constraint for adjacent strips because different temperatures may be mixed in the same plate. It should be noted that while any solution generated by merging the ones obtained by all the subproblems of the first stage is always feasible; without any additional processing it may be a poor quality solution, as many empty strips could exists between temperatures. The objective in this stage will be to rearrange the strips so that all those empty strips will be occupied, leaving empty plates that can be filtered and eliminated. This stage tackles the first objective (filtering the plates) and the third objective (maximizing the occupation of the first plates). However, the second objective cannot be improved in this stage, because the number of control reagents and their locations do not change.

The ILP model that describes this situation (ILP3) is as follows.

The model parameters are obtained from the configuration of the solutions of all the subproblems of the first stage. On the one hand, the set $\{1, \ldots, r\}$ of all the strips of the first stage is considered. For each strip l in this set, the occupation of the strip is known, α_l and its assigned temperature, t_l. On the other hand, the binary parameter z_l reflects whether all the reagent controls are present, whereas the binary parameter δ_l indicates if the strip is empty.

The only variables of the model are the ones that allow to determine the position k in the final solution for each strip l of the first stage, denoted by s_{lk}. Formally:

$$\text{min.} \sum_{q=1}^{p} w_q \sum_{k=6(q-1)+1}^{6q} \sum_{l=1}^{r} \alpha_l \cdot s_{lk}, \qquad w_q = q, \, \forall q \in \{1, \ldots, p\} \qquad (21)$$

subject to:

$$\sum_{l=1}^{r} s_{lk} = 1, \qquad\qquad\qquad \forall k \in \{1, \ldots, 6 \cdot p\} \qquad (22)$$

$$\sum_{k=1}^{6p} s_{lk} = 1, \qquad\qquad\qquad \forall l \in \{1, \ldots, r\} \qquad (23)$$

$$s_{lk} \leq s_{(l-1)(k-1)} + z_l, \qquad \forall l \in \{1, \ldots, r\}, \forall k \in \{1, \ldots, 6 \cdot p\} \qquad (24)$$

$$s_{l(6(q-1)+1)} \leq z_l, \qquad\qquad \forall q \in \{1, \ldots, p\}, \forall l \in \{1, \ldots, r\} \qquad (25)$$

$$\sum_{l=1}^{r} s_{l(k+j)} \cdot t_l - \sum_{l=1}^{r} s_{lk} \cdot t_l \leq 5j + T_{max} \cdot \left(\sum_{l=1}^{r} s_{lk} \cdot \delta_l + \sum_{l=1}^{r} s_{l(k+j)} \cdot \delta_l \right),$$

$$\forall q \in \{1, \ldots, p\}, \forall j \in \{1, \ldots, 5\}, \forall k \in \{6 \cdot (q-1) + 1, \ldots, 6 \cdot q - j\} \qquad (26)$$

$$\sum_{l=1}^{r} s_{lk} \cdot t_l - \sum_{l=1}^{r} s_{l(k+j)} \cdot t_l \leq 5j + T_{max} \cdot \left(\sum_{l=1}^{r} s_{lk} \cdot \delta_l + \sum_{l=1}^{r} s_{l(k+j)} \cdot \delta_l \right),$$

$$\forall q \in \{1, \ldots, p\}, \forall j \in \{1, \ldots, 5\}, \forall k \in \{6 \cdot (q-1) + 1, \ldots, 6 \cdot q - j\} \qquad (27)$$

$$s_{lk} \in \{0,1\}, \qquad\qquad \forall l \in \{1,\dots,r\}, \forall k \in \{1,\dots,6 \cdot p\}$$

It can be seen that the objective function, (21), is defined with a familiar expression, because the idea is still to organize the samples such that the occupation of the first plates is maximized. The difference in this case is that the number of occupied wells is fixed regardless of the operation, because neither the quantity nor the location of the control reagents of the groups will be modified.

Constraints (22) and (23) ensure that each strip is allocated only once and that all strips present in the first stage will be allocated somewhere in the global solution. Constraints (26) and (27) address the temperature difference constraint for adjacent strips. Finally, constraints (24) and (25) ensure that groups fragmented into more than one strip will not be separated into distinct plates along the solution, as that would create feasibility issues.

3.2. Method 2. Heuristic Approach

Unlike M1, this method, known as M2, uses a heuristic algorithm to solve the sub-problems of the first stage. It will be referred to as the Algorithm 2 (A2). However, both methods share the same model, ILP3, as the preferred solver for the second stage.

The starting point for A2 is A1, as presented in Section 2.2, but adapted to this new situation. Specifically, A2 consists of only one movement type, strip exchange, and it is responsible of organizing the solution's strips in the best possible way. The other movement, the sample grouping, has been replaced by a post-processing operation on the plates, whose main function is to optimize the occupation of each plate independently, reducing the fragmentation and dispersion of the samples that make up the groups. In addition, this processing gives greater freedom to the strip exchange movement.

As shown in Algorithm 1, this post-processing operation is executed for each iteration performed by the simulated annealing, that is, each time a complete cooling cycle is made. This gives the algorithm time to explore the solution space and readjust the current solution. Figure 6 shows an example of the process.

Algorithm 1: Simulated annealing with regroup post-process

$currentSol \leftarrow initialSol$
for $i = 0$ **to** $maxIter$ **do**
  $T \leftarrow Tmax$
  while $T \geq Tmin$ **do**
    $newSol \leftarrow stripExchange(currentSol)$
    if $difference(newSol, currentSol) >= 0$ **then**
      $currentSol \leftarrow newSol$
    else
      $threshold \leftarrow e^{\frac{difference}{T}}$
      if $generateRandom() < threshold$ **then**
        $currentSol \leftarrow newSol$
    $T \leftarrow \alpha * T$
  $postProcess(currentSol)$
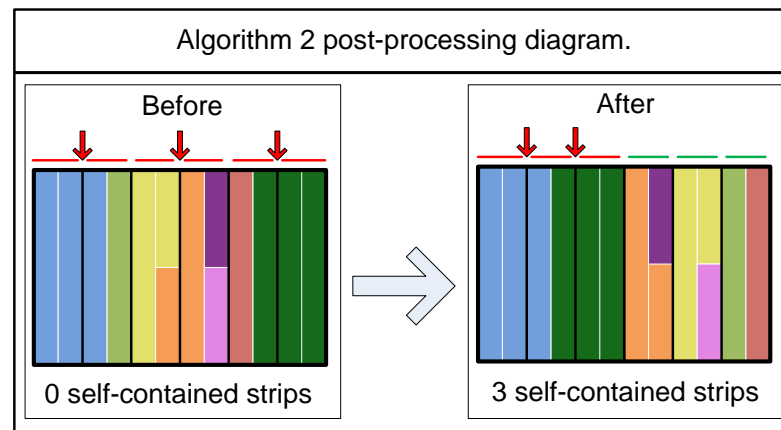**return** $bestSol$

Figure 6. Diagram of post-processing applied to A2. Before processing there is no available strips to exchange (highlighted in red). However, after processing 3 strips are now self-contained and can be moved out (highlighted in green).

Note that each colour represents samples belonging to the same group. As shown, before processing the plate, no individual strip can be moved out because there is no space left for additional control reagents. Red arrows show which groups are present in more than one strip. After processing the plate, several strips are now self-contained, which means that they can be exchanged. Thus, this process provides more freedom to the exchange movement.

To process the plates in each subproblem, the Bin Packing model with item fragmentation as proposed in [11] has been adapted. This process will be referred to as ILP4.

The model is applied to each of the plates separately, taking as parameters the distribution of the samples of the groups present in each strip. Again, the variables v_{il} are considered. The objective is to minimize the dispersion of the groups in strips, so that these can be used by the exchange movement.

$$\text{Min} \sum_{i=1}^{n^j} \sum_{l=1}^{6} v_{il} \tag{28}$$

subject to:

$$\sum_{l=1}^{6} n_{il}^j = N_i, \qquad \forall i \in \{1, \ldots, n^j\} \tag{29}$$

$$\sum_{i=1}^{n} n_{il}^j \leq 16, \qquad \forall l \in \{1, \ldots, 6\} \tag{30}$$

$$\frac{n_{il}^j}{N_i} \leq v_{il}, \qquad \forall i \in \{1, \ldots, n\}, \forall l \in \{1, \ldots, 6\} \tag{31}$$

$$n_{il}^j \in \mathbb{N}, v_{il} \in \{0, 1\} \qquad \forall i \in \{1, \ldots, n\}, \forall l \in \{1, \ldots, 6\}$$

This model is valid for this process because it tackles each plate separately, where all strips share temperature and no control reagent can leave the plate in any way. The model treats all the samples in a group, including the control reagent, as a single fragmentable item to be packed using the strips as containers. The objective function, (28), minimizes the total number of fragments. Constraints (29) and (30) ensure that all samples will be allocated and that the capacity of the strips will not be exceeded. The constraint (31) controls the assignment of groups to strips.

Although this is an exact algorithm based on the model's implementation, the fact that it is executed on individual plates means it has a short execution time, despite the complexity of the problem.

4. Results and Discussion

The goal of this section is to present the results obtained by the models developed in the paper, as well to investigate the impacts of the various improvements proposed for their implementation. Comparisons with other solvers in the literature will also be presented. First, the adopted dataset is analysed to describe its main properties.

4.1. Dataset Analysis

In the study, 30 real daily data files were considered, all of which are included in Section A.1 in the Appendix A. Each data file corresponds to a working session in the laboratory of the company Health in Code. Figure 7 shows a breakdown of these files according to the relevant parameters which include the number of samples, the number of groups and the different processing temperatures. Clearly, the temperature range is limited and does not necessarily depend on the number of samples in the file. In addition, data files were classified into three subsets based on their number of samples. Thus, small files have up to 434 samples, medium files have up to 1128 samples and large files have up to 3783 samples.
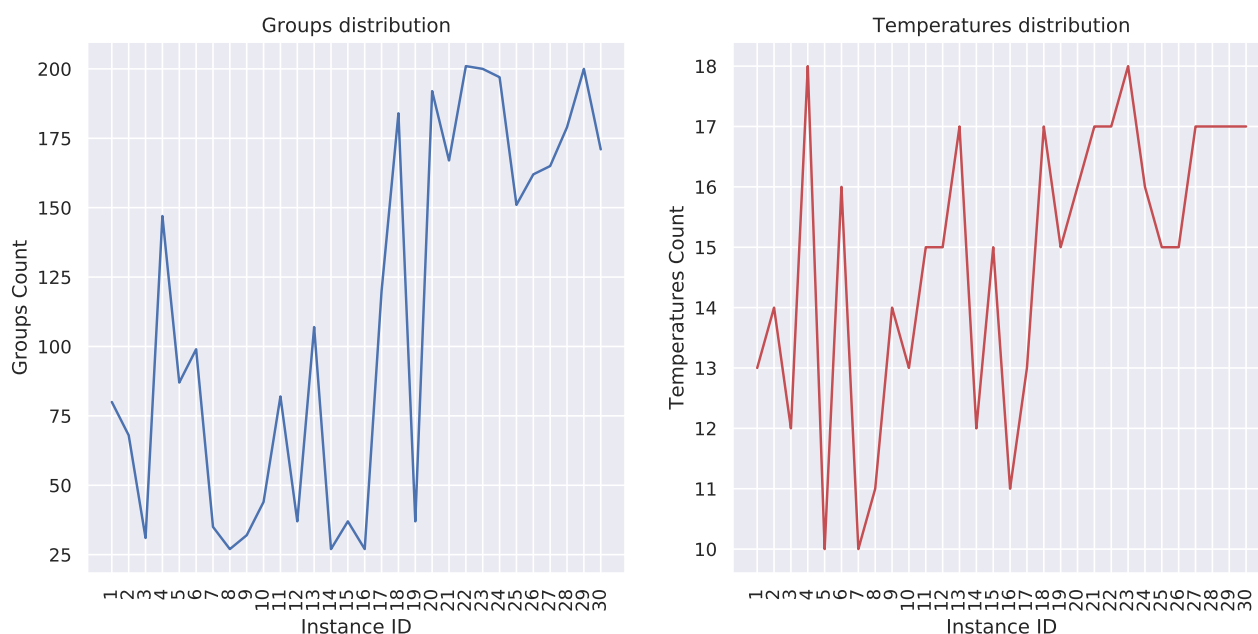
Figure 7. Summary of all 30 available data files.

In Section 3.1.1 it is mentioned that the complexity of the subproblems is highly variable. To illustrate this characteristic, Figure 8 shows the temperature breakdown of a particular file that represents a typical situation in our data files. Two conclusions can be drawn. First, the temperature range is restricted to between 50 °C and 70 °C. Second, due to factors such as the type of experiments performed by this laboratory, where some reagents are widely used, some temperatures are more common than others. An identifier was assigned to each of the data files so that they can be located easily.
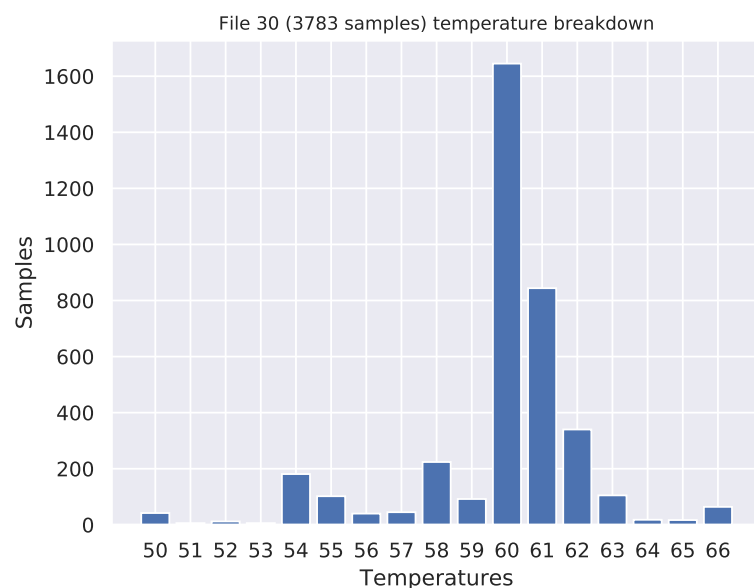
Figure 8. Example of the temperature distribution in a concrete data file.

4.2. Comparison of the Different Approaches

This section is devoted to the results obtained by the methods studied and proposed in the previous section. All of these methods were implemented in Python 3.7. To ensure fair comparisons, all the experiments were executed multiple times for the same duration (600 s) on a single PC equipped with an Intel 8700K CPU, 64 GB RAM and Windows 10. In addition, the ILP models were solved using Gurobi 8.1.

4.2.1. M1 vs. ILP1

To compare these two methods under the same conditions, they were both executed for 600 s and the obtained results were collected. However, ILP1 was unable to obtain a feasible solution within that time for larger files. For this reason, the comparison between M1 and ILP1 involves only the small size files, the largest ones that ILP1 managed to solve within the established time. These solutions are detailed in Appendix A.2 in the Appendix A.

Figure 9 shows a comparison between M1 and ILP1. Note that for each subset of files, all the objectives were compared, namely, the total number of plates used, the total number of occupied wells and the occupation rate of the plates.

Although the solutions achieved by ILP1 are good, M1 improves them consistently. For example, M1 reduces the number of plates in file 18, as well as reduce the total occupation across the entire test suite. Finally, the occupation rate achieved by M1 is better in general terms, filling more plates up to a 100%, especially for larger problems even though M1 uses a fairly lower amount of wells.
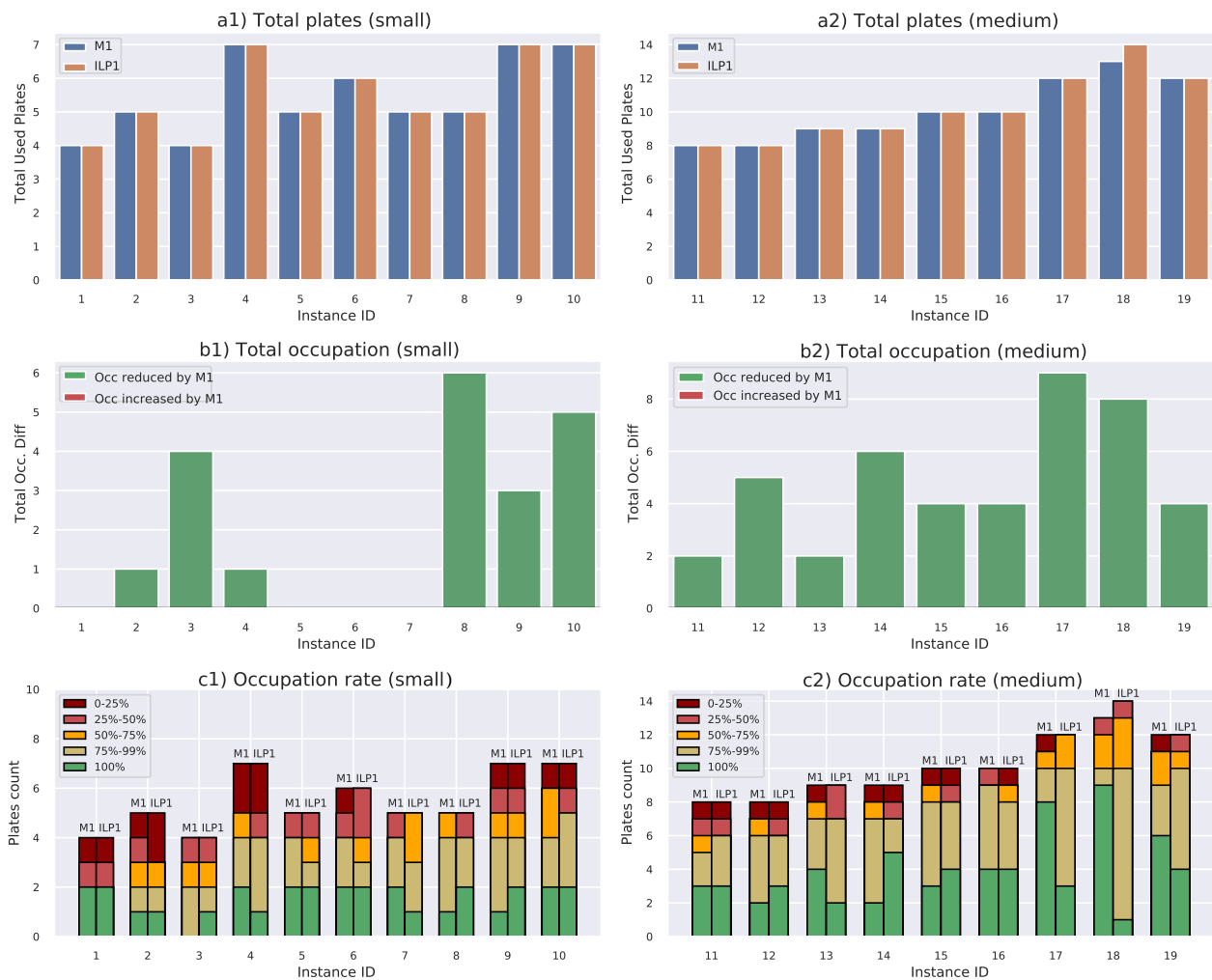
Figure 9. Comparison of M1 and ILP1 on small and medium size files.

4.2.2. M2 vs. A1

Figure 10 shows a comparison between M2 and A1. Clearly, M2 obtains better results. First, it reduces the number of total plates in two instances (files 22 and 30). The total occupation is similar for the smallest files, but the difference becomes greater as the file size increases, eventually, M2 achieves an improvement of up to 50 wells. It should be noted that M2 is able to catch up with plates occupation rate even considering that it occupies a fewer number of wells. In both cases, best solution found is compared, details about best and worst solutions of these methods can be seen in Appendex A.3 and A.5 in the Appendix A. Note that, due to its stability, even the worst solution achieved by M2 is clearly better than the best one of A1.
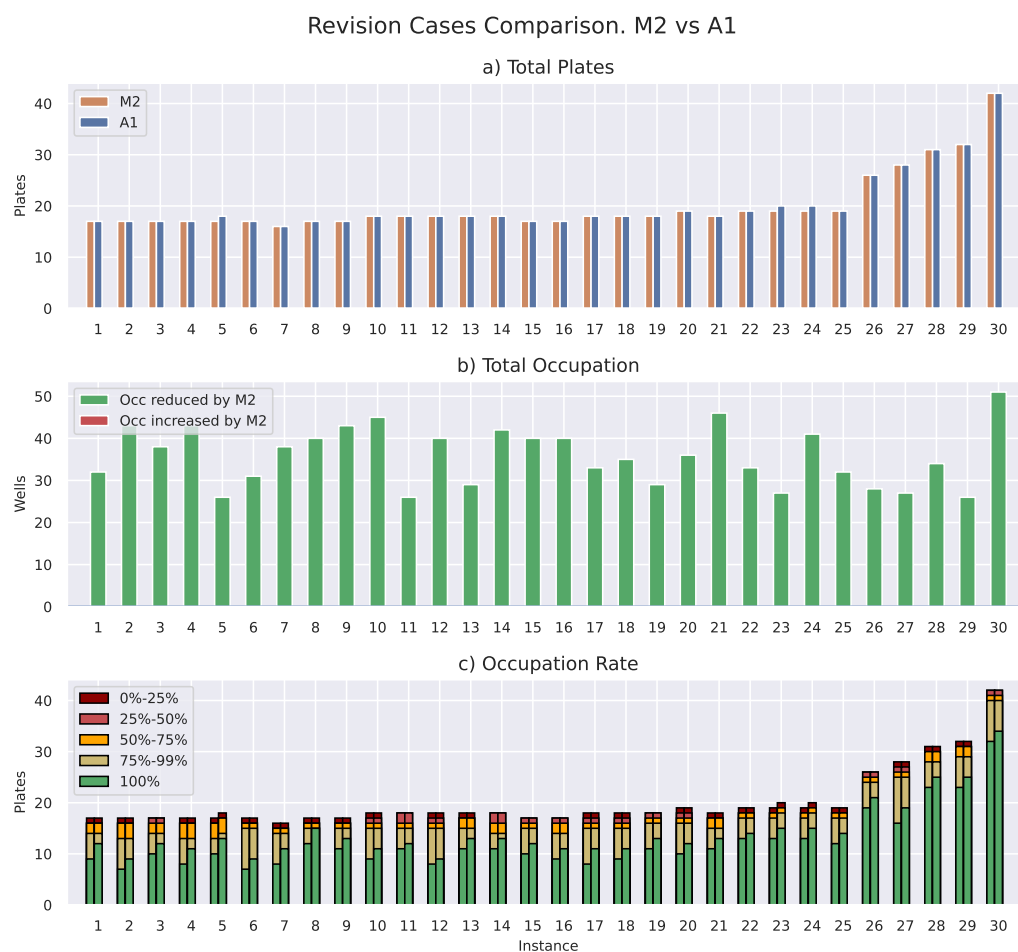
Figure 10. Comparison of M2 and A1 on all available files.

4.2.3. M1 vs. M2

Figure 11 shows a comparison between the two methods proposed in this work, M1 and M2. Both methods achieve solutions of similar quality; in fact, there are no differences in the total number of plates in any file. In addition, the total occupation achieved by both methods is almost identical, the differences amount to only a few wells in favour of the one method or the other. M1 achieves a better occupation rate in general, most notably on the larger files, in fact, it is these files for which M1 reduces the total occupation. The M1 solutions are detailed in Appendix A.4 in the Appendix A.

Furthermore, note that M2's first stage is a heuristic method, which means that it would be desirable to ensure its convergence so far as possible. Figure 12 shows a study of this phenomenon using files 22 and 30 and solving each 20 times. The left part of the figure shows how increasing the execution time ensures that the algorithm reaches an optimal number of plates more times. As shown in the right part of the figure, the total occupation decreases and stabilizes as the execution time increases.

Figure 11. Comparison of M1 and M2 on all available files.

Figure 12. Evolution of M2 solutions over time, files 22 and 30.

4.3. Improving the Performance of the First Stage. Multicore Execution

Because the subproblems of the first stage are independent, several techniques can be explored to minimize their resolution time. Examples of these techniques are listed below and depicted in Figure 13.

Figure 13. Examples of different execution methods.

The independent first stage subproblems allows multicore execution techniques to be explored to minimize their resolution time. In this case, multicore execution means that more than one subproblem can be solved simultaneously, with each of subproblem sharing CPU resources. This is usually accomplished by assigning one CPU core to each task.

In this paper, various multicore implementation techniques have been explored, as summarized below.

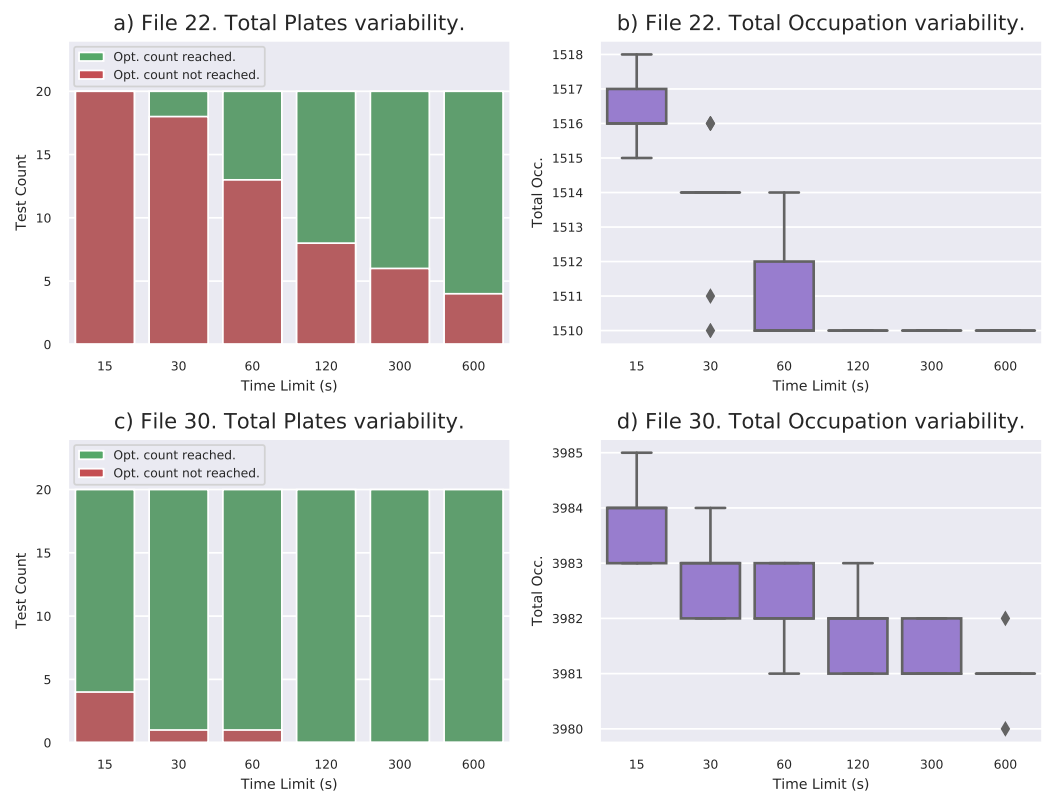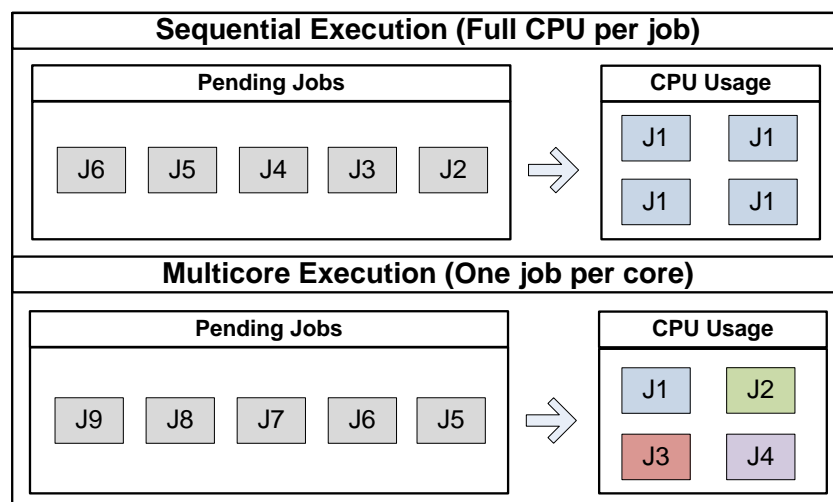- **#Cores**. In this case, it is desirable to instantiate as many processes as there are available cores; thus, each subproblem is assigned to a full core.
- **#Jobs**. In this approach a different process is instantiated for each subproblem, which means that there will be more processes than cores. However, because the majority of the subproblems are solved in seconds, their impact is imperceptible.
- **2Jobs**. Finally, because it have been seen that there are at most two subproblems that are very time consuming, it tries to share resources halfway.

Figure 14 shows a performance comparison of these approaches on the most complex subproblems in file 30. The figure shows that a greater computing capacity results in a better performance; however, the disadvantage is that assigning greater capacity means that fewer subproblems can be solved at once. However, as Table 1 shows, because the majority of these subproblems can be solved in fractions of a second, it is preferable to prioritize the most complex subproblems (the ones with 60 °C and 61 °C).

Table 1. Subproblems detailed for file 30, 3783 samples.

Instance: 3783_260111KAPA35			**Temps: 17**	**Groups: 171**	**Samples: 3783**
Temp.	**Groups**	**Samples**	**Problem Size**	**Time**	**Gap**
50 °C	3	42	75×57	0.0064	0.0
51 °C	1	6	29×19	0.001	0.0
52 °C	2	12	52×38	0.0013	0.0
53 °C	1	6	29×19	0.0011	0.0
54 °C	10	181	462×380	13.8452	0.0
55 °C	10	102	462×380	0.9911	0.0
56 °C	3	40	75×57	0.0019	0.0
57 °C	6	45	144×114	0.0278	0.0
58 °C	13	224	889×741	0.5663	0.0
59 °C	8	92	372×304	2.0234	0.0
60 °C	55	1645	$19{,}511 \times 16{,}720$	4303.7317	0.0299
61 °C	32	844	5712×4864	1618.5057	0.0137
62 °C	15	340	1023×855	5.5391	0.0
63 °C	7	105	327×266	0.0829	0.0
64 °C	1	18	29×19	0.0012	0.0
65 °C	2	17	52×38	0.0019	0.0
66 °C	2	64	52×38	0.0019	0.0
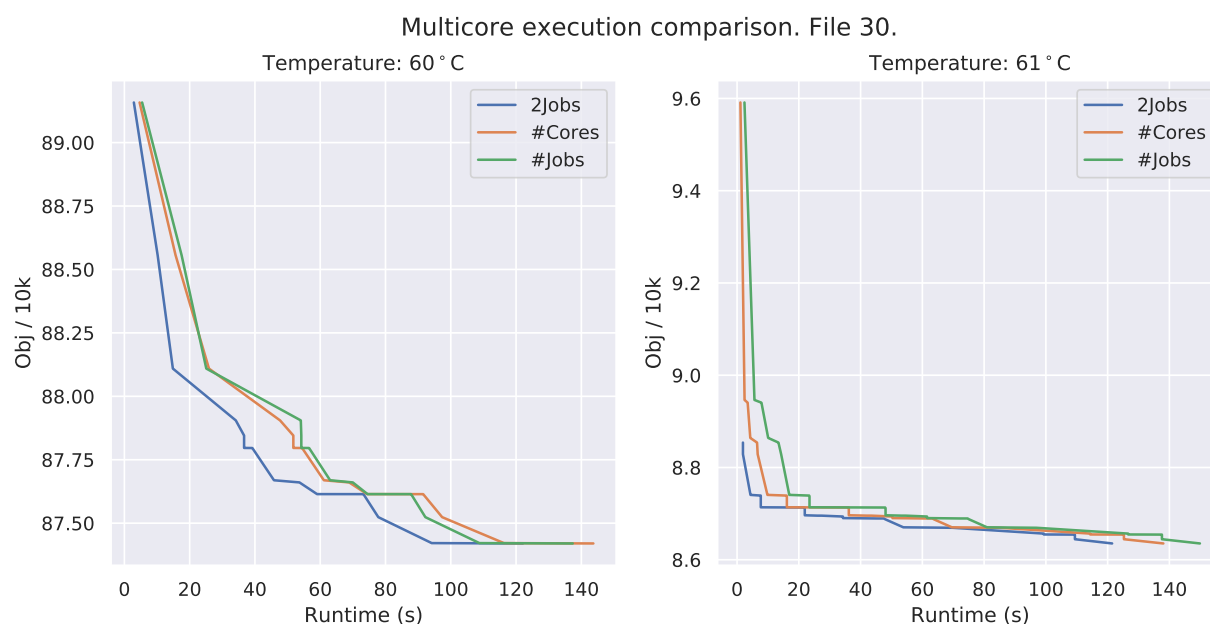
Figure 14. Evolution of the objective function for different multicore approaches.

Multinode Execution

Furthermore, if sufficient computational capacity is available, it is possible to use more than one node, (more than one machine) to solve the subproblems. For example, in an ideal situation where at least one node exists for each job, the total execution time will be the time required by the most challenging subproblem. This situation is depicted in Figure 15. In addition, note that each node resource can be assigned in a multicore configuration if needed.
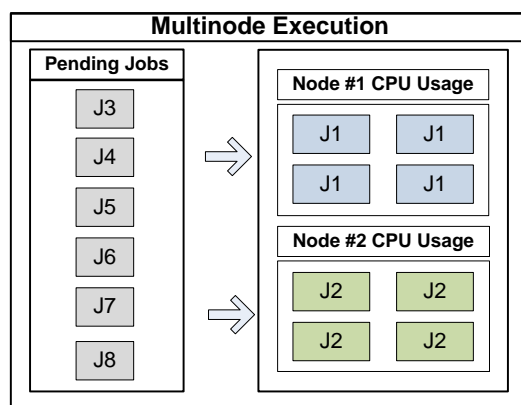
Figure 15. Multinode execution diagram.

To show the potential impact of increasing the number of nodes on the overall performance of the first stage, an experiment was carried out. Two nodes were available with the same characteristics and used to execute the different subproblems of each file. The procedure is as follows: each node is assigned a subproblem to be solved and when that process ends, it reports the solution and is assigned a new one. After this process finishes, the second stage solver is executed to obtain the global solution and the total time consumed is reported. The results are shown in Figure 16, showing that the execution time achieved by the multinode approach is always lower.
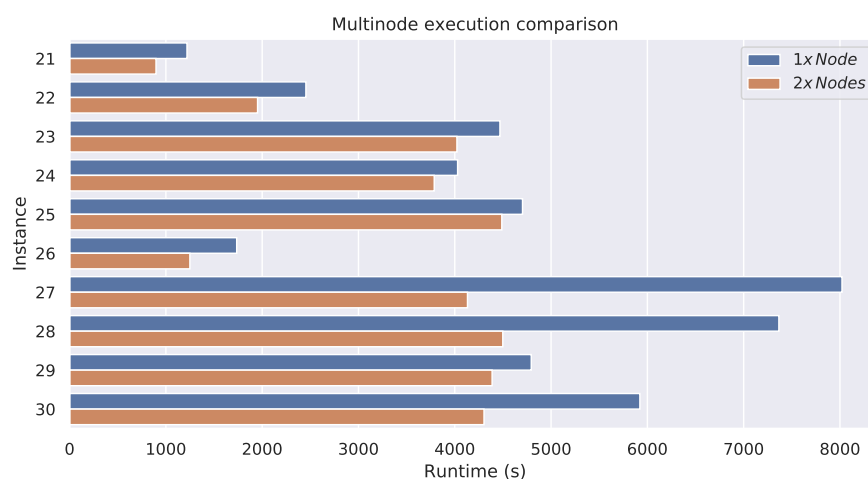
Figure 16. Time consumed by different execution approximations.

4.4. First Stage Impact on the Second Stage

The second stage performance is affected by the quality of the solutions reached in the first stage. Figure 17 shows a comparison of the execution time depending on the method used to solve the first stage. It is easy to see that the model always finds the optimal solution in a reasonable time, consuming approximately 1 min in the worst case for the largest instance. In addition, it should be noted that the solutions provided by M1 generally require less time because M1 is capable of reducing the group fragmentation, which means that the second stage model can pack the strips more efficiently.
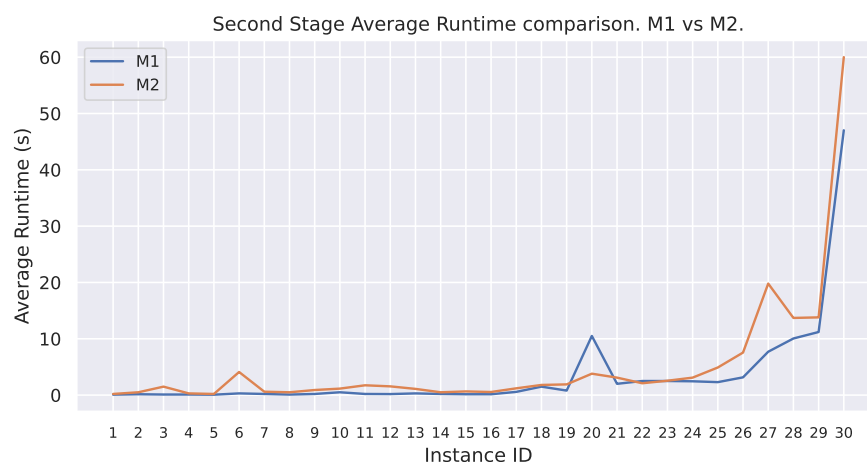
Figure 17. Second stage execution time comparison, depending on the first stage solver.

4.5. Proposed Methods vs. LIMS (Labware)

Labware is a commercial software package used bin the laboratory; therefore, its results define the baseline on against all other proposals should be compared. In this regard, Figure 18 shows a comparison between the methods proposed in this work and Labware. The results of the proposed methods are clearly superior to those provided by Labware on all the proposed objectives. The number of plates is clearly lower in the proposed methods and can even be reduced by 50% in the larger cases. Both methods are capable of reducing the total occupation by tens or even hundreds of samples; however, Labware is not able to optimize the plate occupation percentage, resulting in a greater variability.
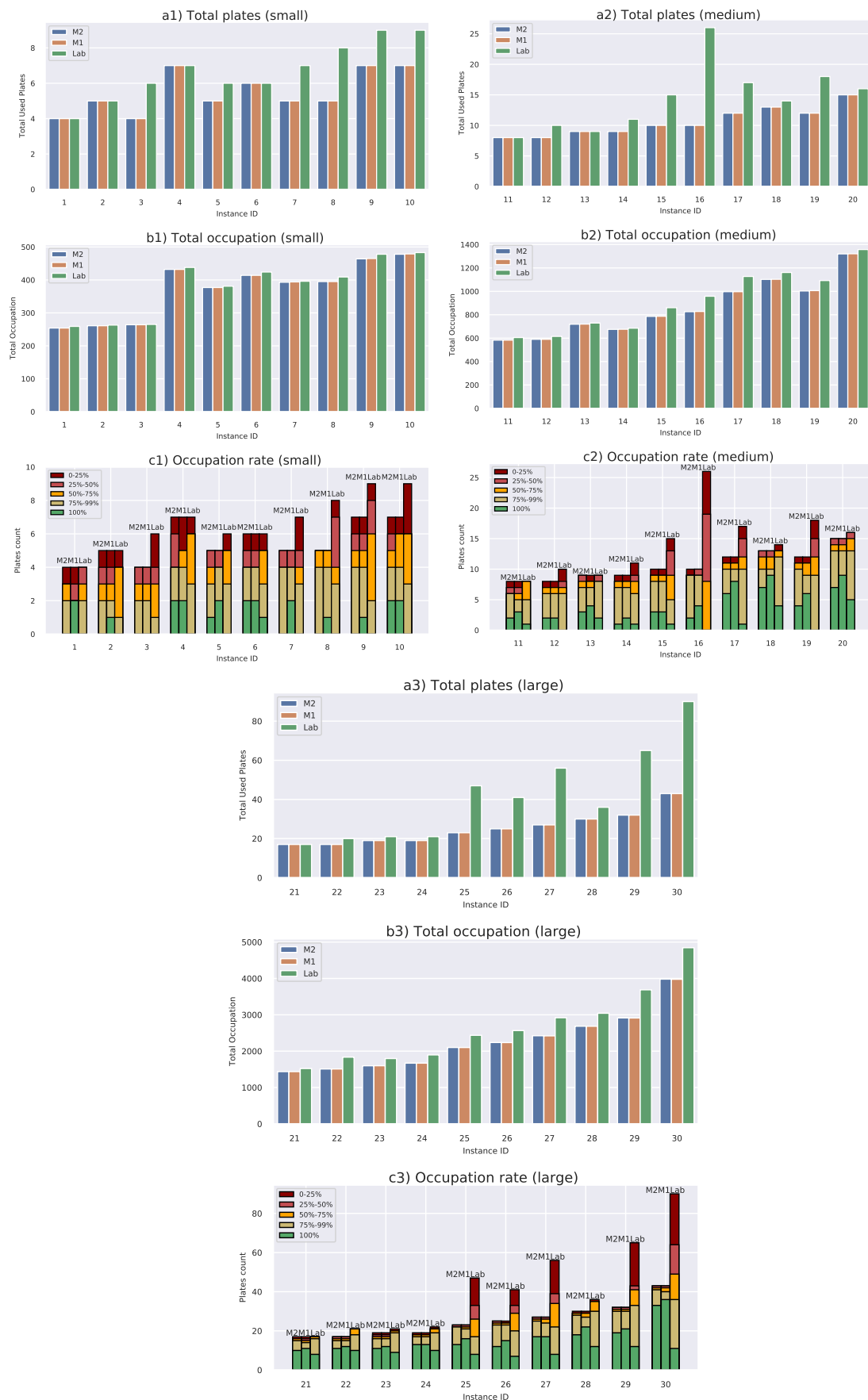
Figure 18. Comparison between M1, M2 and Labware for all available files.

5. Conclusions

This paper deals with a two-stage procedure to efficiently allocate samples in plates as part of the PCR process. First, the inner structure of the problem was studied and how to break it into smaller subproblems that can be solved independently. Thus, numerous simpler problems are tackled instead of a single large one.

To solve the subproblems of the first stage, an ILP model was proposed, being possible to achieve optimal solutions. Furthermore, to avoid potential scalability issues, a heuristic algorithm was also developed, solving the problems at a low computational cost while maintaining the solution quality. For the second stage, an ILP model was developed to merge the solutions of all the subproblems to form a global solution.

According to this procedure, two new methods in two-stages were proposed: M1 (with the ILP problem in the first stage) and M2 (replacing the ILP problem with the heuristic algorithm). Also, multicore and multinode approaches were explored in order to improve the execution times of the first stage subproblems. Notice that this procedure is not effective with the methods developed in [8], since the temperatures are not treated independently.

Overall, three objectives are evaluated, namely, the total number of plates used to allocate all samples and control reagents, the total number of occupied wells in the plates and the occupation rate of the plates. To ensure a fair comparison, all of the methods and algorithms were implemented and executed for the same time span on the same computer. Additionally, the dataset consisted of 30 files of real daily data. It has been shown that both M1 and M2 methods present similar results. Moreover, they have been superior in terms of the quality of the solutions to the software previously used in the laboratory and the heuristic algorithm implemented in [8].

Future lines of research may involve related business challenges. For example, the laboratory may need to establish priorities or time windows for sample processing. On the other hand, if the company migrates to weekly scheduling, it might need dynamic reallocation methods to process samples that arrive on a daily basis. Methods such as M2 could be very useful in these cases.

Author Contributions: Conceptualization, L.C., S.L.-F. and D.N.-D.; methodology, L.C., S.L.-F. and D.N.-D.; software, L.C., S.L.-F. and D.N.-D.; validation, L.C., S.L.-F. and D.N.-D.; writing—original draft preparation, L.C., S.L.-F. and D.N.-D.; writing—review and editing, L.C., S.L.-F. and D.N.-D. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Detailed Results

Appendix A.1. Data Files Detailed

All available and studied files are detailed in Table A1, where relevant features are shown. The number of samples is usually the feature used to classify the potential complexity of the file. The reason is that the number of samples can vary considerably. However, this is not the case of other features, such as the number of different temperatures or the number of groups of samples.

Another critical feature of the files is the distribution of the samples in each temperature. There is great variability in the number of samples processed according to each of the temperatures. In fact, it can range from one sample to hundreds of samples.

Table A1. Daily available files detailed.

ID	Samples	Temps.	Groups	Samples per Temperature
1	174	13	80	[1,1,2,4,5,16,15,74,25,11,15,3,2]
2	193	14	68	[2,7,13,12,1,3,6,23,11,43,21,45,1,5]
3	233	12	31	[24,2,6,1,6,18,36,46,34,29,21,10]
4	285	18	147	[6,15,11,4,20,12,14,18,11,21,90,27,22,9,2,1,1,1]
5	290	10	87	[3,56,24,74,54,9,45,4,16,5]
6	315	16	99	[4,4,4,2,21,7,6,1,8,5,18,87,126,15,6,1]
7	358	10	35	[1,3,3,5,100,116,60,21,7,42]
8	368	11	27	[8,13,6,13,79,51,80,32,46,27,13]
9	432	14	32	[1,1,13,9,8,4,17,85,20,125,41,50,40,18]
10	434	13	44	[2,12,2,16,8,13,83,98,127,22,13,6,32]
11	501	15	82	[8,6,12,6,6,61,2,105,83,175,10,1,23,1,2]
12	551	15	37	[1,1,12,5,12,5,1,22,53,29,174,107,53,58,18]
13	612	17	107	[10,24,40,10,109,41,18,11,125,60,35,96,9,10,10,2,2]
14	647	12	27	[1,27,4,1,2,142,86,168,66,81,48,21]
15	747	15	37	[2,2,12,15,12,6,2,10,55,63,258,160,92,44,14]
16	797	11	27	[36,13,18,6,6,38,204,168,105,147,56]
17	876	13	120	[20,10,30,12,26,111,100,275,136,110,21,3,22]
18	918	17	184	[12,5,14,3,100,59,9,42,72,33,224,113,187,20,2,17,6]
19	963	15	37	[2,2,17,6,24,10,2,44,88,58,290,176,107,101,36]
20	1128	16	192	[17,21,16,2,114,65,23,48,82,37,306,153,199,36,4,5]
21	1270	17	167	[2,8,34,4,16,9,24,41,65,51,461,324,152,43,4,8,24]
22	1309	17	201	[4,14,36,1,28,24,8,39,65,60,436,298,236,46,6,3,5]
23	1398	18	200	[4,6,23,1,27,31,16,35,59,59,480,328,257,56,1,6,3,6]
24	1473	16	197	[1,9,38,34,35,30,38,73,71,520,364,188,50,11,8,3]
25	1944	15	151	[18,3,77,56,40,22,83,99,823,431,180,66,9,8,29]
26	2071	15	162	[25,28,86,64,29,49,177,232,745,339,172,70,6,7,42]
27	2248	17	165	[28,7,9,4,84,53,39,25,154,30,1011,481,219,64,12,8,20]
28	2496	17	179	[9,16,58,9,25,24,52,83,116,93,913,635,304,85,8,16,50]
29	2703	17	200	[24,15,21,7,126,93,48,46,174,72,1090,563,265,85,16,11,47]
30	3783	17	171	[42,6,12,6,181,102,40,45,224,92,1645,844,340,105,18,17,64]

Tables A2–A4 show the Labware solutions in detail for small, medium and large files, respectively.

Table A2. Labware solutions detailed for small files.

	Labware Solution Detail (Small)		
ID	Plates	Occ.	Occupation Rate
1	4	259	[93.75, 81.25, 53.13, 41.67]
2	5	263	[78.13, 72.92, 58.33, 57.29, 7.29]
3	6	265	[76.04, 73.96, 60.42, 40.63, 13.54, 11.46]
4	7	438	[91.67, 84.38, 77.98, 66.67, 66.67, 59.38, 9.38]
5	6	381	[98.96, 85.42, 76.04, 73.96, 57.29, 5.21]
6	6	424	[100.0, 95.83, 95.83, 67.71, 62.50, 19.79]
7	7	396	[97.92, 93.75, 90.63, 59.38, 44.79, 23.96 ,2.08]
8	8	409	[82.29, 80.21, 78.13, 63.54, 48.96, 30.21, 25.00, 17.71]
9	9	478	[91.67, 89.58, 70.83, 61.46, 58.33, 56.25, 33.33, 29.17, 7.29]
10	9	483	[93.75, 91.67, 85.42, 71.88, 60.42, 56.25, 23.96, 17.71, 2.08]

Table A3. Labware solutions detailed for medium files.

	Labware Solution Detail (Medium)		
ID	**Plates**	**Occ.**	**Occupation Rate**
11	8	605	[100.0, 94.79, 85.42, 83.33, 77.08, 71.88, 59.38, 58.33]
12	10	615	[96.88, 89.58, 89.58, 83.33, 83.33, 75.00, 61.46, 34.38, 19.79, 7.29]
13	9	729	[(100.0)×2, 97.92, 90.63, 88.54, 88.54, 80.21, 78.13, 35.42]
14	11	685	[100.0, 95.83, 84.38, 80.21, 78.13, 76.04, 69.79, 55.21, 34.38, 19.79, 19.79]
15	15	860	[100.0, 93.75, 91.67, 86.46, 84.38, 69.79, 64.58, 62.50, 54.17, 47.92, 45.83, 31.25, 28.13, 18.75, 16.67]
16	26	958	[70.83, 67.71, 67.71, 67.71, 60.42, 54.17, 51.04, 50.00, 46.88, 39.58, 39.58, 39.58, 33.33, 31.25, 30.21, 28.13, 28.13, 28.13, 25.00, 23.96, 19.79, 19.79, 18.75, 18.75, 18.75, 18.75]
17	17	1127	[100.0, 98.96, 97.92, 96.88, 94.79, 92.71, 87.50, 87.50, 85.42, 84.38, 67.71, 51.04, 46.88, 36.46, 29.17, 12.50, 4.17]
18	14	1161	[(100.0)×4, 98.96, 95.83, 92.71, 92.71, 90.63, 84.38, 83.33, 82.29, 66.67, 21.88]
19	18	1092	[98.96, 97.92, 96.88, 91.67, 88.54, 85.42, 85.42, 83.33, 76.04, 68.75, 66.67, 50.00, 47.92, 40.63, 25.00, 19.79, 7.29, 7.29]
20	16	1357	[(100.0)×5, 97.92, 97.92, 96.88, 93.75, 91.67, 90.63, 89.58, 82.29, 71.88, 69.79, 31.25]

Table A4. Labware solutions detailed for big files.

	Labware Solution Detail (Large)		
ID	**Plates**	**Occ.**	**Occupation Rate**
21	17	1522.0	[(100.0)×8, (98.96)×2, 92.71, (90.63)×2, 84.38, 83.33, 80.21, 65.63]
22	20	1834.28	[(100.0)×10, 98.96, 94.79, 90.63, (89.58)×2, 87.5, 85.42, 71.88, 68, 75, 59.38]
23	21	1797.0	[(100.0)×9, (98.96)×2, 97.92, 94.79, 93.75, 92.71, 90.63, 86.46, 83.33, 76.04, 54.17, 4.17]
24	21	1897.0	[(100.0)×10, 96.88, (95.83)×2, 94.79, 92.71, 91.67, 88.54, 86.46, 85.42, 70.83, 57.29, 19.79]
25	47	2436.0	[(100.0)×8, (92.71)×2, (90.63)×2, 88.54, 87.5, (83.33)×2, 81.25, 72.92, 70.83, 67.71, 66.67, 61.46, 57.29, 56.25, 54.17, 52.08, 40.63, (36.46)×2, 35.42, 31.25, 30.21, 29.17, 23.96, 19.79, 17.71, (16.67)×2, 15.63, 11.46, (5.21)×2, (4.17)×2, 3.13, (2.08)×2]
26	41	2566.0	[(100.0)×7, (98.96)×2, (97.92)×2, (96.88)×2, 90.63, 89.58, 88.54, 85.42, 83.33, 82.29, 75.0, 72.92, 70.83, (69.79)×2, 67.71, 66.67, 63.54, 62.5, 50.0, 47.92, 44.79, 31.25, 29.17, 16.67, 10.42, 4.17, (3.13)×2, (2.08)×2]
27	56	2919.0	[(100.0)×8, (98.96)×2, 91.67, (88.54)×3, (87.5)×2, 86.46, (81.25)×2, (77.08)×2, 73.96, (67.71)×2, (66.67)×2, 65.63, (58.33)×3, (51.04)×3, 47.92, 42.71, 33.33, 32.29, 29.17, 17.71, 14.58, (7.29)×3, , 6.25, 5.21, (4.17)×3, (3.13)×5, (2.08)×2]
28	36	3042.0	[(100.0)×12, (98.96)×4, 97.92, 96.88, 93.75, 92.71, 90.63, 88.54, (86.46)×2, (85.42)×3, (84.38)×2, 79.17, 72.92, 70.83, 68.75, 67.71, 50.0, 5.21]
29	65	3687.0	[(100.0)×12, (98.96)×2, 97.92, 95.83, 93.75, 91.67, (91.67)×5, 89.58, 86.46, 85.42, 84.38, 83.33, (82.29)×2, 79.17, 78.13, 77.08, 72.92, (68.75)×2, (67.71)×2, 65.63, 64.58, 55.21, 41.67, 40.63, (17.71)×2, (16.67)×2, 15.63, 11.46, 9.38, 7.29, (6.25)×2, (5.21)×2, (4.17)×2, (3.13)×3, (2.08)×5]
30	90	4845.0	[(100.0)×11, 96.88, 94.79, (93.75)×2, 92.71, (91.67)×3, (90.63)×3, 89.58, (88.54)×3, 87.5, (86.46)×4, 85.42, (83.33)×2, 78.13, 77.08, (73.96)×2, (71.88)×3, 70.83, (59.38)×2, 54.17, 52.08, (51.04)×2, 50.0, (48.96)×4, 47.92, 46.88, 42.71, 41.67, 39.58, (38.54)×2, (35.42)×2, 33.33, 27.08, (22.92)×2, 21.88, 18.75, (16.67)×5, 15.63, (14.58)×2, 13.54, (11.46)×2, 7.29, (6.25)×2, 5.21, (4.17)×2, (3.13)×2, (2.08)×2]

Appendix A.2. ILP1 Results Detailed

Tables A5 and A6 show ILP1 solutions in detail for small and medium size files. It can be seen that, as the size of the problem increases, the gap value raises, evidencing the difficulty of obtaining optimal solutions in a reasonable time. Big size files are detailed in Table A7, where no feasible solution has been achieved in the time limit provided.

Table A5. ILP1 results obtained for small size files, limited to 600 s. Runtime column shows time stamp of best solution achieved.

			ILP1 Solution Detail (Small, 600 s)				
Id	Samples	Problem Size	Runtime	Gap	Plates	Occ.	Occupation Rate
1	174	4792 × 4176	231.26 s	0.046	4	254	[100.0, 92.71, 42.71, 29.17]
2	193	5208 × 4530	521.22 s	0.054	5	262	[100.0, 90.62, 58.33, 18.75, 5.21]
3	233	2171 × 1800	458.68 s	0.072	4	268	[100.0, 90.62, 57.29, 30.21]
4	285	14,700 × 13146	586.64 s	0.055	7	433	[100.0, 97.92, 96.88, 79.17, 42.71, 28.12, 6.25]
5	290	6342 × 5550	196.01 s	0.031	5	377	[(100.0)×2, 96.88, 57.29, 38.54]
6	315	8757 × 7740	586.86 s	0.060	6	414	[(100.0)×2, 95.83, 65.62, 35.42, 34.38]
7	358	2910 × 2430	376.67 s	0.072	5	394	[100.0, 96.88, 88.54, 54.17, 70.83]
8	368	2412 × 1980	398.92 s	0.043	5	401	[100.0, 98.96, 97.92, 73.96, 45.83]
9	432	3952 × 3318	432.53 s	0.048	7	468	[(100.0)×2, 98.96, 88.54, 66.67, 26.04, 7.29]
10	434	5014 × 4284	456.97 s	0.059	7	484	[(100.0)×2, 92.71, 90.62, 84.38, 28.12, 7.29]

Table A6. ILP1 results obtained for medium size files, limited to 600 s. Runtime column shows time stamp of best solution achieved.

			ILP1 Solution Detail (Medium, 600 s)				
Id	Samples	Problem Size	Runtime	Gap	Plates	Occ.	Occupation Rate
11	501	9810 × 8640	563.24 s	0.052	8	585	[(100.0)×3, 85.42, 94.79, 80.21, 28.12, 20.83]
12	551	5085 × 4320	595.24 s	0.042	8	595	[(100.0)×3, 98.96, 94.79, 78.12, 34.38, 13.54]
13	612	14,084 × 12,528	581.28 s	0.081	9	722	[100.0, 96.88, 100.0, 94.79, 91.67, 92.71, 47.92, 95.83, 32.29]
14	647	4374 × 3618	516.37 s	0.032	9	682	[(100.0)×4, 97.92, 80.21, 25.0, 7.29]
15	747	6347 × 5400	374.14 s	0.035	10	791	[(100.0)×3, 98.96, 98.96, 97.92, 97.92, 80.21, 43.75, 6.25]
16	797	4797 × 3960	553.65 s	0.028	10	832	[(100.0)×4, 98.96, 98.96, 97.92, 80.21, 72.92, 17.71]
17	876	20,496 × 18,288	523.63 s	0.118	12	1006	[(100.0)×3, 98.96, 96.88, 90.62, 89.58, 88.54, 88.54, 84.38, 53.12, 57.29]
18	918	35,940 × 32,424	499.75 s	0.131	14	1111	[100.0, 98.96, 98.96, 96.88, 95.83, 93.75, 92.71, 90.62, 86.46, 82.29, 66.67, 61.46, 53.12, 39.58]
19	963	7609 × 6480	480.25 s	0.039	12	1010	[(100.0)×4, 98.96, 98.96, 96.88, 91.67, 91.67, 89.58, 56.25, 28.12]

Table A7. ILP1 results obtained for big size files, limited to 600 s. No solutions have been found in time.

Id	Samples	Problem Size	Runtime	Gap	Plates	Occ.	Occupation Rate
20	1128	39,972 × 36,090	-	-	-	-	-
21	1270	39,828 × 35,904	-	-	-	-	-
22	1309	50,151 × 45,360	-	-	-	-	-
23	1398	52,792 × 47,766	-	-	-	-	-
24	1473	51,820 × 46,854	-	-	-	-	-
25	1944	48,750 × 43,884	-	-	-	-	-
26	2071	56,562 × 51,000	-	-	-	-	-
27	2248	62,454 × 56,376	-	-	-	-	-
28	2496	74,849 × 67,680	-	-	-	-	-
29	2703	88,584 × 80,256	-	-	-	-	-
30	3783	105,111 × 95,040	-	-	-	-	-

Appendix A.3. A1 Results Detailed

This section presents the results obtained by A1 for all sizes, detailing the objectives and other relevant information. Best and worst solutions achieved for small size files are

shown in Table A8. It is easy to see that A1 is not able to always obtain the same solution in the established time. In larger files, such as the ones shown in Tables A9 and A10, the behaviour is similar. In any case, the most important objective, namely, the total number of plates, is always the same.

It should be noted that differences between solutions are present in the total number of occupied wells and the occupation rate of the plates, where better solutions reduce the number of occupied wells, thus reducing the count of used control reagents. Improving the occupation rate of the plates by itself does not improve the quality of a solution, because this improvement may be caused by the use of larger amounts of control reagents, which worsens the solution as a whole.

Table A8. A1 best and worst solutions obtained for small files, limited to 600 s.

A1 Solution Detail (Small, 600 s)						
	Best Solution			**Worst Solution**		
ID	**Plates**	**Occ.**	**Occupation Rate**	**Plates**	**Occ.**	**Occupation Rate**
1	4	254	[100.0, 91.67, 52.08, 20.83]	4	256	[100.0, 89.58, 56.25, 20.83]
2	5	261	[100.0, 83.33, 48.96, 35.42, 4.17]	5	267	[100.0, 82.29, 57.29, 34.38, 4.17]
3	4	265	[91.67, 77.08, 68.75, 38.54]	4	268	[97.92, 70.83, 68.75, 41.67]
4	7	432	[100.0, 97.92, 80.21, 75.0, 51.04, 35.42, 10.42]	7	435	[100.0, 97.92, 85.42, 62.5, 55.21, 47.92, 4.17]
5	5	377	[(100.0)×2, 91.67, 68.75, 32.29]	5	379	[(100.0)×2, 90.62, 66.67, 37.5]
6	6	417	[(100.0)×2, 94.79, 57.29, 56.25, 26.04]	6	420	[(100.0)×2, 97.92, 81.25, 41.67, 16.67]
7	5	398	[(100.0)×2, 95.83, 80.21, 38.54]	5	406	[(100.0)×2, 98.96, 82.29, 41.67]
8	5	403	[(100.0)×3, 75.0, 44.79]	5	410	[(100.0)×3, 73.96, 53.12]
9	7	474	[(100.0)×3, 98.96, 61.46, 23.96, 9.38]	7	479	[(100.0)×3, 97.92, 60.42, 30.21, 10.42]
10	7	492	[(100.0)×3, 95.83, 67.71, 34.38, 14.58]	7	497	[(100.0)×3, 96.88, 73.96, 37.5, 9.38]

Table A9. A1 best and worst solutions obtained for medium files, limited to 600 s.

A1 Solution Detail (Medium, 600 s)						
	Best Solution			**Worst Solution**		
ID	**Plates**	**Occ.**	**Occupation Rate**	**Plates**	**Occ.**	**Occupation Rate**
11	8	596	[(100.0)×4, 97.92, 75.0, 42.71, 5.21]	8	599	[(100.0)×5, 78.12, 40.62, 5.21]
12	8	600	[(100.0)×5, 73.96, 37.5, 13.54]	8	608	[(100.0)×5, 73.96, 43.75, 15.62]
13	9	731	[(100.0)×4, 96.88, 86.46, 81.25, 62.5, 34.38]	9	738	[(100.0)×4, 96.88, 89.58, 81.25, 68.75, 32.29]
14	9	697	[(100.0)×6, 79.17, 37.5, 9.38]	9	703	[(100.0)×6, 85.42, 35.42, 11.46]
15	10	804	[(100.0)×6, 98.96, 78.12, 45.83, 14.58]	10	813	[(100.0)×6, 98.96, 82.29, 44.79, 20.83]
16	10	847	[(100.0)×7, 90.62, 68.75, 22.92]	10	853	[(100.0)×7, 93.75, 65.62, 29.17]
17	12	1019	[(100.0)×8, 98.96, 91.67, 58.33, 12.5]	12	1023	[(100.0)×8, 98.96, 76.04, 73.96, 16.67]
18	13	1119	[(100.0)×7, 98.96, 98.96, 95.83, 78.12, 50.0, 43.75]	13	1124	[(100.0)×9, 95.83, 67.71, 58.33, 48.96]
19	12	1030	[(100.0)×9, 92.71, 46.88, 33.33]	12	1037	[(100.0)×9, 89.58, 54.17, 36.46]
20	15	1364	[(100.0)×11, 95.83, 94.79, 72.92, 57.29]	15	1371	[(100.0)×12, 94.79, 72.92, 60.42]

Table A10. A1 best and worst solutions obtained for large files, limited to 600 s.

			A1 Solution Detail (Large, 600 s)				
		Best Solution				Worst Solution	
ID	Plates	Occ.	Occupation Rate	Plates	Occ.	Occupation Rate	
21	17	1484	[(100.0)×12, 96.88, 94.79, 70.83, 64.58, 18.75]	17	1490	[(100.0)×12, 98.96, 88.54, 65.62, 53.12, 45.83]	
22	18	1544	[(100.0)×13, 96.88, 91.67, 67.71, 38.54, 13.54]	18	1548	[(100.0)×12, 98.96, 96.88, 90.62, 57.29, 56.25, 12.5]	
23	19	1630	[(100.0)×13, 96.88, 91.67, 76.04, 60.42, 52.08, 20.83]	19	1632	[(100.0)×13, 93.75, 76.04, 70.83, 59.38, 54.17, 45.83]	
24	19	1702	[(100.0)×14, 98.96, 97.92, 69.79, 62.5, 43.75]	19	1706	[(100.0)×14, 97.92, 85.42, 82.29, 69.79, 41.67]	
25	23	2129	[(100.0)×18, 91.67, 91.67, 87.5, 76.04, 70.83]	23	2132	[(100.0)×19, 94.79, 90.62, 81.25, 54.17]	
26	25	2271	[(100.0)×20, 95.83, 91.67, 85.42, 59.38, 33.33]	25	2285	[(100.0)×19, 98.96, 97.92, 92.71, 83.33, 70.83, 36.46]	
27	27	2471	[(100.0)×22, 97.92, 87.5, 84.38, 69.79, 34.38]	27	2476	[(100.0)×22, 97.92, 96.88, 95.83, 61.46, 27.08]	
28	30	2713	[(100.0)×23, 95.83, 93.75, 89.58, 86.46, 64.58, 62.5, 33.33]	30	2732	[(100.0)×23, 98.96, 97.92, 92.71, 79.17, 67.71, 66.67, 42.71]	
29	32	2960	[(100.0)×26, 97.92, 90.62, 89.58, 73.96, 72.92, 58.33]	32	2966	[(100.0)×27, 94.79, 90.62, 82.29, 69.79, 52.08]	
30	44	4023	[(100.0)×38, 89.58, 87.5, 85.42, 62.5, 61.46, 4.17]	44	4034	[(100.0)×37, 97.92, 94.79, 83.33, 76.04, 59.38, 50.0, 40.62]	

Appendix A.4. M1 Results Detailed

Detailed results for M1 are shown in Tables A11–A13.

Table A11. M1 solutions detailed for small files, limited to 600 s.

		M1 Solution Detail (Small, 600 s)	
ID	Plates	Occ.	Occupation Rate
1	4	254	[(100.0)×2, 43.75, 20.83]
2	5	261	[100.0, 85.42, 56.25, 28.12, 2.08]
3	4	264	[91.67, 75.0, 72.92, 35.42]
4	7	432	[(100.0)×2, 94.79, 78.12, 55.21, 17.71, 4.17]
5	5	377	[(100.0)×2, 92.71, 75.0, 25.0]
6	6	414	[(100.0)×2, 96.88, 86.46, 39.58, 8.33]
7	5	394	[(100.0)×2, 85.42, 81.25, 43.75]
8	5	395	[100.0, 87.5, 86.46, 80.21, 57.29]
9	7	465	[100.0, 94.79, 93.75, 81.25, 70.83, 39.58, 4.17]
10	7	479	(100.0)×2, 89.58, 88.54, 63.54, 51.04, 6.25]

Table A12. M1 solutions detailed for medium files, limited to 600 s.

		M1 Solution Detail (Medium, 600 s)	
ID	Plates	Occ.	Occupation Rate
11	8	583	[(100.0)×3, 97.92, 94.79, 73.96, 35.42, 5.21]
12	8	590	[(100.0)×2, 97.92, 94.79, 85.42, 77.08, 53.12, 6.25]
13	9	720	[(100.0)×4, 98.96, 92.71, 78.12, 65.62, 14.58]
14	9	676	[(100.0)×2, 98.96, 87.5, 84.38, 83.33, 80.21, 62.5, 7.29]
15	10	787	[(100.0)×3, 98.96, 97.92, 90.62, 84.38, 83.33, 58.33, 6.25]
16	10	828	[(100.0)×4, 98.96, 92.71, 88.54, 79.17, 78.12, 25.0]
17	12	997	[(100.0)×8, 95.83, 80.21, 57.29, 5.21]
18	13	1103	[(100.0)×9, 91.67, 76.04, 58.33, 22.92]
19	12	1006	[(100.0)×6, 97.92, 97.92, 95.83, 71.88, 70.83, 13.54]
20	15	1321	[(100.0)×9, 98.96, 94.79, 88.54, 81.25, 66.67, 45.83]

Table A13. M1 solutions detailed for large files, limited to 600 s.

	M1 Solution Detail (Large, 600 s)		
ID	**Plates**	**Occ.**	**Occupation Rate**
21	17	1437	[(100.0)×11, 96.88, 90.62, 86.46, 71.88, 46.88, 4.17]
22	17	1510	[(100.0)×12, 98.96, 93.75, 83.33, 58.33, 38.54]
23	19	1599	[(100.0)×12, 98.96, 94.79, 90.62, 81.25, 64.58, 33.33, 2.08]
24	19	1670	[(100.0)×13, 97.92, 94.79, 88.54, 78.12, 56.25, 23.96]
25	23	2100	[(100.0)×16, 98.96, 98.96, 98.96, 95.83, 88.54, 69.79, 36.46]
26	25	2236	[(100.0)×15, 98.96, 98.96, 98.96, 97.92, 95.83, 94.79, 90.62, 77.08, 55.21, 20.83]
27	27	2421	[(100.0)×17, 98.96, 98.96, 98.96, 96.88, 95.83, 89.58, 84.38, 73.96, 59.38, 25.0]
28	30	2685	[(100.0)×22, 97.92, 96.88, 95.83, 94.79, 83.33, 71.88, 51.04, 5.21]
29	32	2912	[(100.0)×21, 98.96, 98.96, 97.92, 96.88, 95.83, 90.62, 85.42, 83.33, 78.12, 67.71, 39.58]
30	43	3977	[(100.0)×36, 97.92, 95.83, 90.62, 85.42, 73.96, 65.62, 33.33]

Appendix A.5. M2 Results Detailed

In terms of comparison between best and worst solutions, M2 behaviour is quite different from A1. For small size problems, detailed in Table A14, both solutions are identical, which means that M2 converges. However, as seen with A1, as the size of the files increase, gaps between solutions begin to appear. However, unlike A1, M2 is able to reduce the variability of the solutions. For example, as shown in Table A15, in medium size files there is only one occupied well of difference between best and worst solution. In addition, occupation rate is almost identical in any case. Lastly, for the largest files, detailed in Table A16, variability is also small, greater than in medium size files, but overall there is a gap of less than 10 wells between best and worst solution.

Table A14. M2 best and worst solutions obtained for small and medium files, limited to 600 s.

	M2 Solution Detail (Small, 600 s)						
	Best Solution				**Worst Solution**		
ID	**Plates**	**Occ.**	**Occupation Rate**	**Plates**	**Occ.**	**Occupation Rate**	
1	4	254	[100.0, 93.75, 48.96, 21.88]	4	254	[100.0, 93.75, 48.96, 21.88]	
2	5	261	[96.88, 81.25, 60.42, 31.25, 2.08]	5	261	[96.88, 81.25, 60.42, 31.25, 2.08]	
3	4	264	[92.71, 79.17, 70.83, 32.29]	4	264	[92.71, 79.17, 70.83, 32.29]	
4	7	432	[(100.0)×2, 92.71, 80.21, 47.92, 25.0, 4.17]	7	432	[(100.0)×2, 92.71, 80.21, 47.92, 25.0, 4.17]	
5	5	377	[100.0, 96.88, 86.46, 72.92, 36.46]	5	377	[100.0, 96.88, 86.46, 72.92, 36.46]	
6	6	414	[(100.0)×2, 96.88, 86.46, 39.58, 8.33]	6	414	[(100.0)×2, 96.88, 86.46, 39.58, 8.33]	
7	5	393	[98.96, 95.83, 93.75, 81.25, 39.58]	5	393	[98.96, 95.83, 93.75, 81.25, 39.58]	
8	5	395	[98.96, 91.67, 84.38, 75.0, 61.46]	5	395	[98.96, 91.67, 84.38, 75.0, 61.46]	
9	7	464	[97.92, 97.92, 92.71, 85.42, 69.79, 37.5, 2.08]	7	464	[97.92, 97.92, 92.71, 85.42, 69.79, 37.5, 2.08]	
10	7	478	[100.0, 97.92, 94.79, 85.42, 70.83, 44.79, 4.17]	7	478	[100.0, 97.92, 94.79, 85.42, 70.83, 44.79, 4.17]	

Table A15. M2 best and worst solutions obtained for small and medium files, limited to 600 s.

	M2 Solution Detail (Medium, 600 s)					
	Best Solution				**Worst Solution**	
ID	Plates	Occ.	Occupation Rate	Plates	Occ.	Occupation Rate
11	8	583	[(100.0)×3, 96.88, 93.75, 73.96, 38.54, 4.17]	8	583	[(100.0)×3, 96.88, 93.75, 73.96, 38.54, 4.17]
12	8	590	[(100.0)×2, 98.96, 93.75, 87.5, 78.12, 52.08, 4.17]	8	590	[(100.0)×2, 98.96, 93.75, 87.5, 78.12, 52.08, 4.17]
13	9	719	[(100.0)×3, 93.75, 89.58, 87.5, 82.29, 69.79, 26.04]	9	719	[(100.0)×3, 93.75, 89.58, 87.5, 82.29, 69.79, 26.04]
14	9	675	[100.0, 92.71, 92.71, 87.5, 87.5, 86.46, 84.38, 67.71, 4.17]	9	675	[100.0, 92.71, 92.71, 87.5, 87.5, 86.46, 84.38, 67.71, 4.17]
15	10	786	[(100.0)×3, 95.83, 94.79, 90.62, 90.62, 87.5, 53.12, 6.25]	10	787	[(100.0)×4, 94.79, 90.62, 90.62, 84.38, 53.12, 6.25]
16	10	826	[(100.0)×2, 98.96, 96.88, 95.83, 94.79, 90.62, 82.29, 79.17, 21.88]	10	827	[(100.0)×3, 98.96, 95.83, 94.79, 88.54, 82.29, 79.17, 21.88]
17	12	997	[(100.0)×7, 95.83, 87.5, 84.38, 65.62, 5.21]	12	997	[(100.0)×7, 95.83, 87.5, 84.38, 65.62, 5.21]
18	13	1102	[(100.0)×6, 98.96, 96.88, 94.79, 90.62, 76.04, 59.38, 31.25]	13	1102	[(100.0)×6, 98.96, 96.88, 94.79, 90.62, 76.04, 59.38, 31.25]
19	12	1003	[(100.0)×4, 96.88, 94.79, 91.67, 91.67, 90.62, 87.5, 69.79, 21.88]	12	1004	[(100.0)×4, 96.88, 95.83, 94.79, 94.79, 89.58, 87.5, 69.79, 16.67]
20	15	1320	[(100.0)×8, 97.92, 96.88, 93.75, 92.71, 84.38, 68.75, 40.62]	15	1320	[(100.0)×8, 97.92, 96.88, 93.75, 92.71, 84.38, 68.75, 40.62]

Table A16. M2 best and worst solutions obtained for large files, limited to 600 s.

	M2 Solution Detail (Large, 600 s)					
	Best Solution				**Worst Solution**	
ID	Plates	Occ.	Occupation Rate	Plates	Occ.	Occupation Rate
21	17	1437	[(100.0)×9, 98.96, 94.79, 93.75, 88.54, 86.46, 76.04, 54.17, 4.17]	17	1442	[(100.0)×10, 97.92, 93.75, 92.71, 85.42, 76.04, 52.08, 4.17]
22	17	1510	[(100.0)×11, 98.96, 93.75, 89.58, 83.33, 72.92, 34.38]	18	1511	[(100.0)×9, 98.96, 96.88, 94.79, 93.75, 87.5, 83.33, 75.0, 41.67, 2.08]
23	19	1598	[(100.0)×11, 98.96, 93.75, 92.71, 88.54, 83.33, 68.75, 36.46, 2.08]	19	1601	[(100.0)×11, 98.96, 96.88, 93.75, 89.58, 82.29, 67.71, 36.46, 2.08]
24	19	1670	[(100.0)×13, 96.88, 93.75, 89.58, 78.12, 64.58, 16.67]	19	1678	[(100.0)×13, 97.92, 95.83, 92.71, 78.12, 64.58, 18.75]
25	23	2100	[(100.0)×14, 98.96, 98.96, 96.88, 95.83, 93.75, 93.75, 91.67, 80.21, 37.5]	23	2107	[(100.0)×15, 98.96, 98.96, 96.88, 94.79, 93.75, 92.71, 81.25, 37.5]
26	25	2238	[(100.0)×12, 98.96, 98.96, 98.96, 98.96, 98.96, 97.92, 93.75, 93.75, 91.67, 88.54, 81.25, 65.62, 23.96]	25	2241	[(100.0)×13, 98.96, 98.96, 98.96, 97.92, 96.88, 94.79, 93.75, 92.71, 88.54, 81.25, 65.62, 26.04]
27	27	2424	[(100.0)×17, 98.96, 96.88, 96.88, 93.75, 92.71, 90.62, 87.5, 81.25, 59.38, 27.08]	27	2428	[(100.0)×18, 98.96, 96.88, 96.88, 93.75, 90.62, 87.5, 76.04, 59.38, 27.08]
28	30	2685	[(100.0)×16, 98.96, 98.96, 97.92, 96.88, 96.88, 96.88, 95.83, 93.75, 93.75, 93.75, 88.54, 77.08, 59.38, 8.33]	30	2689	[(100.0)×17, 98.96, 98.96, 98.96, 98.96, 98.96, 97.92, 94.79, 93.75, 91.67, 84.38, 77.08, 58.33, 8.33]
29	32	2914	[(100.0)×19, 98.96, 98.96, 97.92, 97.92, 97.92, 95.83, 94.79, 94.79, 91.67, 86.46, 81.25, 67.71, 31.25]	32	2920	[(100.0)×22, 98.96, 98.96, 98.96, 96.88, 94.79, 90.62, 85.42, 80.21, 65.62, 31.25]
30	43	3981	[(100.0)×33, 98.96, 97.92, 97.92, 96.88, 94.79, 91.67, 87.5, 83.33, 65.62, 32.29]	44	3983	[(100.0)×34, 97.92, 97.92, 89.58, 87.5, 85.42, 84.38, 78.12, 71.88, 53.12, 3.12]

References

1. Matt, C.; Hess, T.; Benlian, A.L. Digital Transformation Strategies. *Bus. Inf. Syst. Eng.* **2015**, *57*, 339–343. [CrossRef]
2. Reis, J.; Amorim, M.; Melão, N.; Matos, P. Digital transformation: A literature review and guidelines for future research. In *World Conference on Information Systems and Technologies*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 411–421.
3. Henriette, E.; Feki, M.; Boughzala, I. The shape of digital transformation: A systematic literature review. In Proceedings of the 9th Mediterranean Conference on Information Systems, Samos, Greece, 3–5 October 2015; Volume 10.
4. Herrmann, M.; Boehme, P.; Mondritzki, T.; Ehlers, J.P.; Kavadias, S.; Truebel, H. Digital transformation and disruption of the health care sector: internet-based observational study. *J. Med. Internet Res.* **2018**, *20*, e9498. [CrossRef] [PubMed]
5. van Merode, G.G.; Oosten, M.; Vrieze, O.J.; Derks, J.; Hasman, A. Optimisation of the structure of the clinical laboratory. *Eur. J. Oper. Res.* **1998**, *105*, 308–316. [CrossRef]
6. Turkcan, A.; Zeng, B.; Muthuraman, K.; Lawley, M. Sequential clinical scheduling with service criteria. *Eur. J. Oper. Res.* **2011**, *214*, 780–795. [CrossRef]
7. Sanger, F.; Nicklen, S.; Coulson, A.R. DNA sequencing with chain-terminating inhibitors. *Proc. Natl. Acad. Sci. USA* **1977**, *74*, 5463–5467. [CrossRef] [PubMed]
8. Carpente, L.; Cerdeira-Pena, A.; Lorenzo-Freire, S.; Places, A.S. Optimization in Sanger Sequencing. *Comput. Oper. Res.* **2019**, *109*, 250–262. [CrossRef]
9. Sweeney, P.E.; Paternoster, E.R. Cutting and packing problems: A categorized, application-orientated research bibliography. *J. Oper. Res. Soc.* **1992**, *4*, 691–706. [CrossRef]
10. Delorme, M.; Iori, M.; Martello, S. Bin packing and cutting stock problems: Mathematical models and exact algorithms. *Eur. J. Oper. Res.* **2016**, *255*, 1–20. [CrossRef]
11. LeCun, B.; Mautor, T.; Quessette, F.; Weisser, M.A. Bin packing with fragmentable items: Presentation and approximations. *Theor. Comput. Sci.* **2015**, *602*, 50–59. [CrossRef]
12. Kirkpatrick, S.; Gelatt, C.D., Jr.; Vecchi, M.P. Optimization by simulated annealing. *Science* **1983**, *220*, 671–680. [CrossRef] [PubMed]