The Bin Packing Problem with Fragile Objects: Models and Solution Methods

by

Michelle Bo Qi

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Management Sciences

Waterloo, Ontario, Canada, 2015

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Michelle Bo Qi

Abstract

The Bin Packing Problem (BPP) is an important optimization problem with many applications. Given a set of items with a certain weight and a set of bins with fixed capacity, the classical BPP seeks to pack the items into the minimum number of bins. In this thesis a variant of the BPP, the Bin Packing Problem with Fragile Objects (BPPFO), is studied. The BPPFO originates in telecommunication systems where cellphone calls are assigned to towers based on the noise level of calls and the tolerance level of each call in the channel.In this case, the calls are represented as objects that are characterized by a weight and a fragility parameter. The fragility parameter corresponds to the noise tolerance level of each call, and the weight corresponds to the noise produced by a call. As calls are assigned, the total noise produced within a channel can not exceed the lowest tolerance level among the calls. The less the tolerance level, the more fragile the line becomes. Hence, the capacity of the bin depends on the smallest fragility of any item packed in it. In this thesis, two new formulations are proposed. The first is based on the classical BPP formulation to which a Lagrangian relaxation is applied. Several heuristics are developed to find upper bounds, including a greedy heuristic. The second is a graph-based formulation that is solved directly. A standard data set is used for testing. It is found that the new formulation based on the classical BPP is more efficient in getting a Lagrangian lower bound and the greedy heuristic outperforms other heuristics in finding good quality upper bounds in very short computational times, especially with very large data instances.

Acknowledgments

I would like to express my most sincere gratitude to my supervisor and mentor, Professor Samir Elhedhli, for his support, guidance, and patience throughout my research work. It has been a great honor to have worked under his supervision.

I would also like to thank the readers, Professor James Bookbinder and Professor David Fuller, for their valuable feedback, and for their time put into reviewing my thesis.

Also, thanks to founders of the lab Professor Samir Elhedhli and Professor Fatma Gzara for building such a wonderful place to be working in. Thanks also go to other members of the WatOpt Lab, Ahmed Saif, Da Lu, Yan Wang, Yifeng Yan and my husband, Shou, for their support and help during working together in the lab.

Dedication

I dedicated this thesis to my dear parents, husband, Shuo, and my loving furry family members, Zedong, Pudding, little B, Mimi and Dandan.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The bin-packing problem is a common problem encountered in many applications. The most common application is probably in freight transportation. There are a number of parcels to be delivered, and there are a number of available containers. Trying to fit all the parcels in the least number of containers could be modelled as bin-packing problem. Another application is the layout of a digital circuit on a chip, where different rectangular cells are to be placed on a rectangular shaped chip. The bin-packing problem represents a group of problems like the ones discussed above. It is also being categorized based on certain characteristics of the problem. For example, the freight example is a 3-dimensional bin-packing problem whereas the layout of a chip is a 2-dimensional problem. In the layout problem, the cells may have different purposes and need to be placed in a certain order; this would be the bin-packing problem with priority. In this thesis, we will discuss a particular variant of the bin packing problem, which is the bin-packing problem with fragile objects. Given a set of items characterized by a weight parameter, and large number of bins with fixed capacity, packing the items into minimum number of bins is called the bin-packing problem, BPP in short. The one particular variant we are interested in is the bin-packing problem with fragile objects. In telecommunication systems, calls need to be assigned to a channel with certain frequency bandwidth. Each call has a certain level of noise, and the quality of a call is related to the tolerance of a channel towards total noises produced by all calls in the channel. Therefore, calls need to be assigned in a way such that the total noise in the channel does not exceed the smallest tolerance of any call in this channel. In this case, the calls are represented as objects that are characterized by a weight and a fragility parameter. The fragility parameter corresponds to the noise tolerance level of each call, and the weight corresponds to the noise produced by a call. As calls are assigned, the total weight produced by all calls within a channel can not exceed the lowest fragility level among the calls. The lower the tolerance level, the more fragile the line becomes. Hence, the capacity of the bin depends on the smallest fragility of any item packed in it.This leads to the Bin Packing Problem with Fragile Objects (BPPFO), where the objects to be packed are characterized by a weight and a fragility parameter. The total weight of the items in a bin cannot exceed the lowest fragility of an object in the bin. Figure 1.1 is an example of the problem. [7] The length of the white portion of the bar represents weight, and the total length of the bar is fragility. An optimal solution is given.

In this thesis, Lagrangian relaxation is applied to two integer linear programming formulations. One is a compact model developed by Clautiaux et al. [7], and another is a new model developed based on the classical BPP model. The differences in the subproblems of the two formulations are identified, and the latter

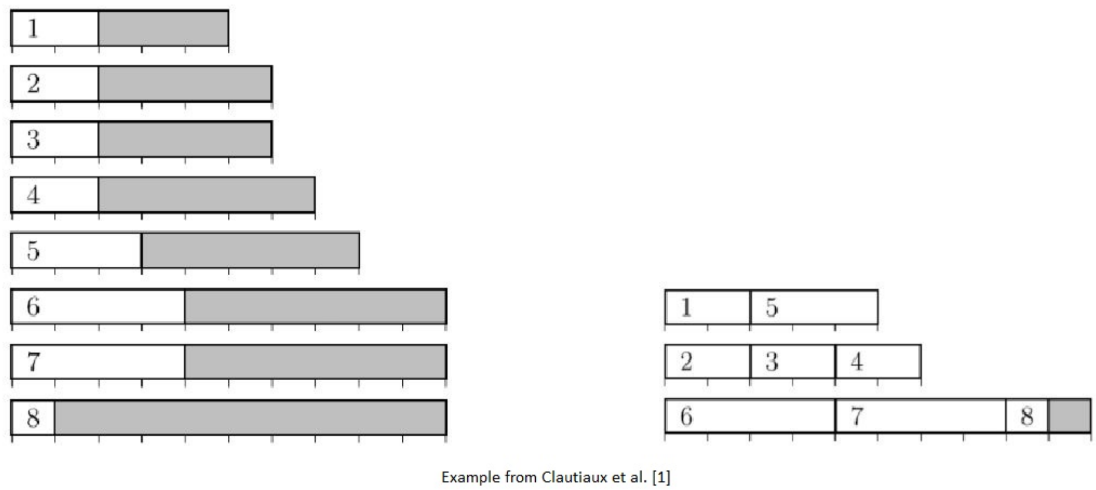


Figure 1.1: An example of the BPPFO

is proven to be more efficient as it only requires the solution of one subproblem, instead of multiple subproblems for the compact model.

A greedy heuristic is also developed to find a feasible solution efficiently. The greedy heuristic is shown to find good quality solutions and even reach optimality in a very short amount of time compared to other heuristics or algorithms. A brand new formulation to model the BPPFO is also presented. A graph is used to model the problem, where items are represented as nodes, and solving for paths through the graph would give the packing of the bins.

The rest of the thesis is outlined as follows: Chapter 2 is a survey of past works on BPP and other variants of it including BPPFO. Chapter 3 details the Lagrangian relaxation on the two different models. Chapter 4 lists the different heuristics used to find upper bounds. Chapter 5 illustrates the graphical approach to the problem. Chapter 6 gives details of the numerical testing and summaries of the numerical results. Finally, Chapter 7 provides a conclusion. The detailed testing results are listed in the Appendix.

Chapter 2

Literature Review

A survey of past sample work on the Bin Packing Problem, its variants and applications including Bin Packing with Fragile Objects will be presented in this chapter.

2.1 Bin Packing Problems and Applications

The Bin Packing Problem is one of the most studied problems in combinatorial optimization. Goodman, Tetelbaum, and Kureichik [12] classified the BPP into three categories, 1 dimensional (1-D), 2 dimensional (2-D), and 3 dimensional (3-D) bin packing, where the dimensions of items and bins differ. This section will review the three categories of the BPP.

2.1.1 The 1-D Bin Packing Problem

The 1-D BPP addresses only one property of the packing. It is used to optimize over one single bin parameter. The 1-D BPP does not model many applications, as in practice, most of the BPP are in 2-D or 3-D. However, it serves as means to develop algorithms to tackle such NP-hard problem, and lays foundation for further studies on more practical applications. It is also used to study other constrained BPP, which have additional practical constraints based on application.

For instance, Berky [3] presented a massive parallel computing method that uses multiprocessor to solve the 1-D BPP. Later, Berky and Wang [4] further developed a systolic parallel approximation algorithm for the 1-D BPP and made improvement on the results. Boutevin [5] relates the bin packing problem to an industrial line balancing problem, and solved it with a generalized next-fit and immediate-update-first-fit algorithms. Quiroz-Castellanos et al. [20] proposed a method called Grouping Genetic Algorithms with Controlled Genes Transmission for BPP that improves over the standard genetic algorithms.

There are other variants for the 1-D BPP. M. Baldi et al. [1] studied the variable cost and size BPP with optional items. The items are characterized by volume and profit, and the bins are of different types characterized by volume and cost. The objective is to minimize the cost of bins while maximizing the total profit of items. An exact approach, branch-and-price, and a heuristic, beam search, are presented.

2.1.2 The 2-D Bin Packing Problem

The 2-D bin packing problem has a wider range of applications. Hong et al. [13] proposed a hybrid heuristic in solving a 2-D variable sized BPP. The hybrid heuristic based on simulated annealing and binary search is used to improve the result of a backtracking algorithm, which utilizes a mixed bin packing algorithm that solves a single bin. The results are proven to outperform other existing algorithms for this problem. Esbensen [10] presented an algorithm based on genetic algorithms and simulated annealing to solve the placement of macro-cell in very-large-scale-integration layout. The problem involves placing rectangular cells on the layout without overlapping and subject to side constraints. The optimization algorithm executes the genetic algorithm or the simulated annealing, or a mixture of both depending on the control parameter. It was proven that the algorithm is very adaptive and yields better results than pure genetic algorithm.

Not only rectangular cases were considered, but also packing circular shaped objects, pipes, polygonal shaped objects into rectangular bins were studied. George et al [11] developed an algorithm to pack different sized circles into a rectangular container. He modelled the problem as a non-linear mixed integer problem and developed various heuristics to solve it. He found that a quasi random technique and genetic algorithm performed best for circle packing problems. Jakobs [15] studied placing polygons in rectangular bin using deterministic packing algorithms. It was found that genetic algorithm can improve any deterministic algorithm and help escape local minima.

2.1.3 The 3-D Bin Packing Problem

The 3-D bin packing problem considers all three dimensions namely length, width and height of items and bins, and has many practical applications.

Hifi et al. [14] proposed two integer linear programming heuristics to solve the 3-D single bin packing problem. The integer linear program is first decomposed into several 3-D knapsack problems. One of the heuristics executes in two phases to pack the bins, while the other extended a local reoptimization phase on top of the first heuristic. Both heuristics are proven to produce good results compared to other literatures. Martello et al. [18] studied the problem to pack a set of rectangular boxes orthogonally into minimum number of 3 dimensional bins. He presented an exact branch-and-bound algorithm for packing a single bin. It solved problems with up to 90 items where many are solved to optimality within reasonable time. Karmakar et al. [16] used a 3-D bin packing model to solve test schedule problem for a System-on-Chip. The test windows are selected with particle swarm optimization, and the 3-D bin packing problem is used to solve for scheduling the test windows.

2.2 Variants of The Bin Packing Problem

There are lots of variants to the bin packing problem. They are studied to address extensions of the bin packing problem for different applications.

2.2.1 The Bin Packing Problem with Conflicts

Another popular variant of the BPP is the Bin Packing Problem with Conflicts (BPC). It is a bin packing problem where conflicting items cannot be placed in the same bin.

Elhedhli et al. [9] proposed a branch-and-price algorithm where the subproblems are knapsack problems with conflicts. Sadykov et al. [21] presents a generic branch-and-price algorithms using different pricing technique to solve the BPC. A dynamic algorithm is developed for when the conflict graph is in interval, and a depth first search pricing for no special structure graph. The solution method updated the benchmark results for this problem with reasonable runtime. Variants of the BPP can also be in higher dimensional case. Khanafer [17] presented a tree-decomposition heuristic to solve a 2-D BPC. The heuristic decomposes the 2-D BPC into subproblems and solves them independently.

2.2.2 The Bin Packing Problem with Precedence Constraints

Another variant that received more attention in recent years is the Bin Packing Problem with Precedence Constraints (BPP-P). Its objective is to minimize the number of bins used to pack items in an order such that certain precedences are satisfied. It has particular applications in assembly and scheduling issues. Dell'Amico et al. [8] first address the BPP-P with an exact solution approach. The approach consists of large number of lower bounds, upper bound from variable neighbourhood search, and a branch-and-bound algorithm. The approach is found to be effective compared to other integer linear programming techniques.

2.3 The Bin Packing Problem with Fragile Objects

In previous work done by Clautiaux et al. [7] and Martinez et al. [19], different lower bounding and upper bounding techniques were explored. These include simple combinatorial lower bounds, column generation; and various heuristics for upper bound. Exact solution methods were also sought.

A simple lower bound proposed by Clautiaux et al. [7] is obtained by simply relaxing the fragility constraint, thus converting the problem into a classical bin packing problem with fixed capacity. The capacity of the bin is set to be the maximum fragility value, so that all items can be packed. The lower bound, L_0, is given by:

$$L_0 = \lceil \sum_{j=1}^{n} \frac{w_j}{f_{max}} \rceil. \tag{2.1}$$

where $w_j, j = 1, \ldots, n$ are item weights and f_{max} is the maximum fragility over all items. It was then improved by Chan et al. [6] who proved the following is a valid lower bound:

$$L_1 = \lceil \sum_{j=1}^{n} \frac{w_j}{f_j} \rceil. \tag{2.2}$$

A better combinatorial lower bound L_2 is presented by Bansal el al. [2] that uses the idea of fractional lower bound. It packs the items in non-decreasing fragility order, and only pack a portion of an item when it does not fit in a bin entirely. The remaining portion is then packed in a newly opened bin.

Clautiaux et al. [7] proposed a complex lower bound based on column generation. As for upper bound, a large set of greedy heuristics were derived from the BPP and the Vertex Coloring literature to fit the BPPFO constraints. A Variable Neighbourhood Search was also developed to find an upper bound. It is found to outperform the other heuristics, especially for lager set of items.

As for exact method for the problem, Martinez et al. [19] proposed a branch-and-bound and several branch-and-price algorithms, which utilize lower bounds and tailored optimization techniques to improve convergence.

Chapter 3

Lagrangian Lower Bounding

In this chapter, Lagrangian relaxation is applied to two different formulations. The first is the one proposed by Clautiaux et al. [7], and the second is a new formulation. The difference of the subproblems will be looked at, and the bound quality will be compared.

3.1 Problem formulation and Lagrangian Relaxation

In this section, we present the compact model proposed by Clautiaux et al.[7] and perform Lagrangian relaxation on one of the constraints to obtain a lower bound.

3.1.1 Problem Formulation

Clautiaux et al. [7] model the problem as an integer program. They use the term "witness" to refer to the item with the smallest fragility in a bin. A binary variable y_i, $i = 1, \ldots, n$ is defined to take value 1 if item i is a witness, and 0 otherwise. Binary variables x_{ij}, $i = 1, \ldots, n, j = i+1, \ldots, n$ are defined to take value 1 if item j is assigned to the bin with item i as the witness, and 0 otherwise. As all items are ordered in non-decreasing order by fragility, only variable x_{ij} where $j = i + 1, ..., n$ are defined. The BPPFO can be modelled as:

$$
\begin{aligned}
[P1]: \quad & \min \quad \sum_{i=1}^{n} y_i && && (3.1)\\
& \text{s.t.} \quad y_j + \sum_{i=1}^{j-1} x_{ij} = 1 && j = 1, \ldots, n && (3.2)\\
& \sum_{j=i+1}^{n} w_j x_{ij} \le (f_i - w_i) y_i && i = 1, \ldots, n && (3.3)\\
& x_{ij} \le y_i && i = 1, \ldots, n, j = i+1, \ldots, n && (3.4)\\
& y_i \in \{0, 1\} && i = 1, \ldots, n && (3.5)\\
& x_{ij} \in \{0, 1\} && i = 1, \ldots, n, j = i+1, \ldots, n && (3.6)
\end{aligned}
$$

The objective (3.1) minimizes the number of witnesses, which is equivalent to minimizing the number of bins used. Constraints (3.2) restrict each item to be packed exactly once, either as a witness or packed in any bin. Constraints (3.3) ensure that the sum of weights in the bin does not exceed the fragility of the witness and constraints (3.4) tighten the linear relaxation. Constraints (3.2) will be relaxed in a Lagrangian fashion.

3.1.2 Lagrangian Relaxation

Applying Lagrangian relaxation on the assignment constraints (3.2) leads to the subproblem

$$
\begin{aligned}
[KP1]: \quad & \sum_{i=1}^{n} \lambda_i + \min \quad \sum_{i=1}^{n}(1-\lambda_i)y_i + \sum_{i=1}^{n}\sum_{j=1}^{i-1}(-\lambda_i x_{ij}) \\
& \qquad \text{s.t.} \quad \sum_{j=i+1}^{n} w_i x_{ji} \le (f_i - w_i)y_i \qquad i = 1, \ldots, n \\
& \qquad\qquad x_{ij} \le y_i \quad i = 1 \ldots n, j = i+1, i+2, \ldots, n \\
& \qquad\qquad y_i \in \{0,1\} \\
& \qquad\qquad x_{ij} \in \{0,1\}
\end{aligned}
\tag{3.7}
$$

which can be decomposed into n 0-1 knapsack problems with fragile objects:

$$
\begin{aligned}
[KP1_i]: \quad & v_i = \min_{x,y} \quad (1-\lambda_i)y_i - \sum_{j=1}^{i-1} \lambda_i x_{ij} \\
& \qquad \text{s.t.} \quad \sum_{j=i+1}^{n} w_i x_{ji} \le (f_i - w_i)y_i \qquad i = 1, \ldots, n \\
& \qquad\qquad x_{ij} \le y_i \quad i = 1 \ldots n, j = i+1, i+2, \ldots, n \\
& \qquad\qquad y_i \in \{0,1\} \\
& \qquad\qquad x \in \{0,1\}
\end{aligned}
\tag{3.8}
$$

The Lagrangian bound LB_{lag1} is $\sum_{i=1}^{n} \lambda_i + \sum_{i=1}^{n} v_i$.

The best bound is the solution of the Lagrangian dual problem:

$\underset{\lambda_i}{max}\left\{ \quad \sum_{i=1}^{n} \lambda_i + v_i \right\}$ which is equivalent to:

$$\begin{aligned} \max \quad & \sum_{i=1}^{n} \lambda_i + \sum_{i=1}^{n} \theta_i \\ \text{s.t.} \quad & z_i + (y_i^h + \sum_{j=1}^{i-1} x_{ij}^h)\lambda_i \le y_i^h \qquad \forall i, h \\ & h = 1, \dots, H \end{aligned} \tag{3.9}$$

where h is the index of feasible solution x_{ji}^h, y_i^h to the subproblem. The dual of (3.9) is the Dantzig-Wolfe master problem of the BPPFO:

$$\begin{aligned} \min \quad & \sum_{i=1}^{n} \sum_{h=1}^{H} y_i^h \alpha_{hi} \\ \text{s.t.} \quad & \sum_{h=1}^{H} (\sum_{j=1}^{h} x_{ij}^h + y_i^h)\alpha_{hi} = 1 \qquad i = 1, \dots, n \\ & \sum_{h=1}^{H} \alpha_{hi} = 1 \qquad i = 1, \dots, n \\ & \alpha_{hi} \ge 0, h = 1 \dots H, i = 1 \dots, n. \end{aligned} \tag{3.10}$$

3.2 A New Formulation

In this section, we provide a new formulation to the problem that is more closely related to the BPP. We will also show the advantage of this model over the model of Clautiaux et al. [7]

3.2.1 Problem Formulation

Another way to model the BPPFO is to closely relate it to the BPP. Given n items $i = 1, \dots, n$ each with weight w_i and fragility f_i, and n bins $j = 1, \dots, n$, we define binary variable z_j that takes value 1 if bin j is used and 0 otherwise. We also define a binary variable x_{ji} for $i = 1, \ldots, n$ and $j = i, \ldots, n$ that takes value 1 if item i is packed in bin j, 0 otherwise. Note that x_{ji} is not defined for $j > i$ because the items are ordered in non-decreasing order of fragility. The BPPFO can then be modelled as:

$$[P2]: \quad \min \quad \sum_{j=1}^{n} z_j \tag{3.11}$$

$$\text{s.t.} \quad \sum_{j=1}^{i} x_{ji} = 1 \qquad \forall i \tag{3.12}$$

$$\sum_{i=\bar{i}}^{n} w_i x_{ji} \leq f_{\bar{i}} x_{j\bar{i}} + \mathrm{M}(1 - x_{j\bar{i}}) \quad j = 1, ..., \bar{i}, \bar{i}, ..., n \tag{3.13}$$

$$x_{ji} \leq z_j \qquad i = 1, \ldots, n, j = 1, \ldots, i \tag{3.14}$$

$$x_{ji}, z_j \in \{0, 1\} \qquad i = 1, ..., n, , j = 1, ..., i \tag{3.15}$$

The objective (3.11) minimizes the number of bins used. Constraints (3.12) restrict each item to be packed exactly once. Constraints (3.13) says when $x_{ji} = 1$, for every item packed in a bin, the sum of weights in the bin should not exceed its fragility; when $x_{ji} = 0$, constraints (3.13) becomes redundant. M is a big number. By rewriting constraints (3.13) and rearranging the variables to the left side, we get

$$\sum_{i=\bar{i}+1}^{n} w_i x_{ji} + [w_{\bar{i}} + \mathrm{M} - f_{\bar{i}}] x_{j\bar{i}} \leq \mathrm{M} \tag{3.16}$$

where M is set to be the maximum fragility. Constraint (3.14) ensures what only open bins are used.

3.2.2 Lagrangian Relaxation

Applying Lagrangian relaxation on the assignment constraint (3.12) leads to the subproblem:

$$
\begin{aligned}
[KP2] \quad \sum_{i=1}^{L} \lambda_i + \min \quad & \sum_{j=1}^{n} z_j - \sum_{j=1}^{i} \sum_{i=1}^{n} \lambda_i x_{ji} \\
\text{s.t.} \quad & \sum_{i=\bar{i}}^{n} w_i x_{ji} \leq f_{\bar{i}} x_{j\bar{i}} + M(1 - x_{j\bar{i}}) \; j = 1, ..., \bar{i}, \bar{i} = 1, ..., n \\
& x_{ji} \leq z_j \qquad i = 1, ..., n, j = i, ..., n \\
& x_{ji}, z_j \in \{0, 1\} \qquad i = 1, ..., n, j = i, ..., n
\end{aligned}
\tag{3.17}
$$

which decomposes into n identical 0-1 knapsack problems with fragile objects, independent of the bin index j:

$$
\begin{aligned}
[KP2] \quad v_{KP2} = \quad \min \quad & z - \sum_{i=1}^{n} \lambda_i x_i \\
\text{s.t.} \quad & \sum_{i=\bar{i}}^{n} w_i x_i \leq f_{\bar{i}} x_{\bar{i}} + M(1 - x_{\bar{i}}) \quad \bar{i} = 1, ..., n \\
& x_i \leq z \qquad i = 1, ..., n \\
& x_i, z \in \{0, 1\} \qquad i = 1, ..., n
\end{aligned}
\tag{3.18}
$$

The Lagrangian bound LB_{lag2} is given for a set of λ by $\sum_{i=1}^{n} \lambda_i + n v_{KP2}$.

The best bound is the solution of the Lagrangian dual problem:

$\underset{\lambda_i}{max} \left\{ \quad \sum_{i=1}^{n} \lambda_i + n v_{KP2} \right\}$ which is equivalent to:

$$\max \quad \sum_{i=1}^{n} \lambda_i + n\theta \tag{3.19}$$

$$\text{s.t.} \quad \theta + \sum_{i=1}^{n} x_i^h \lambda_i \le z^h \quad \forall h \tag{3.20}$$

$$h = 1, \dots, H \tag{3.21}$$

where h is the index of feasible solution x_i^h, z^h to subproblem [KP2]. Considering constraint (3.20), two cases are possible, $z^h = 1$ and $z^h = 0$, which leads to:

$$\begin{aligned}
\max \quad & \sum_{i=1}^{n} \lambda_i + n\theta \\
\text{s.t.} \quad & \theta + \sum_{i=1}^{n} x_i^h \lambda_i \le 1 \quad \forall h \\
& \theta \le 0 \\
& h = 1, \dots, H
\end{aligned} \tag{3.22}$$

The dual of (3.19) is the Dantzig-Wolfe master problem:

$$\min \quad \sum_{h=1}^{H} z^h \alpha_h \tag{3.23}$$

$$\text{s.t.} \quad \sum_{h=1}^{H} x_i^h \alpha_h = 1 \qquad i = 1, ..., n \tag{3.24}$$

$$\sum_{h=1}^{H} \alpha_h \le n \tag{3.25}$$

$$\alpha_h \ge 0, h = 1, ..., H. \tag{3.26}$$

Constraint (3.25) is redundant since n is always greater than or equal to 1, therefore it can be removed. Taking the dual of the Dantzig-Wolfe master problem again gives a new Lagrangian dual problem:

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} \lambda_i \\
\text{s.t.} \quad & \sum_{i=1}^{n} x_i^h \lambda_i \leq 1 \quad \forall h \\
& h = 1, \ldots, H
\end{aligned}
\tag{3.27}
$$

3.3 Conclusion

As seen above, relaxing the assignment constraint in [$P2$] leads to n identical knapsack subproblems, which only needs to be solved once. Whereas relaxing the assignment constraint in [$P1$] leads to multiple knapsack subproblems that need to be solved multiple times. In practice, this would save a lot of computing time. In both models, the same constraint is relaxed, the assignment constraint, which makes sure each item is only assigned once.

Chapter 4

Upper Bounding Methods

In this chapter, we present several heuristics to compute the upper bound. The first heuristic is derived from the subproblem of the Lagrangian algorithm of problem formulation 1. The second utilizes the concept of fixing the witness, and uses Cplex to solve the knapsack problem that follows. The third is a tabu search heuristic, which aims to find a certain sequence of the items, in which they can be packed.

4.1 A Lagrangian Heuristic

The solution obtained by solving $[KP1]$ is often infeasible to the original BPPFO. To get a feasible solution, the subproblem solution, x_{ij}^h and y_j^h are modified to force feasibility to the original BPPFO. The incumbent is updated if the heuristic finds a better solution.

The basic idea of the heuristic is to use what the current subproblem solution provides, detect what is not feasible in the solution and modify it so that it becomes feasible. The steps of the heuristic are:

1. Check for feasibility. If feasible, exit, update incumbent accordingly; if not, continue;

2. Keep a list of witnesses from the solution, and a list of unpacked objects;

3. Copy feasible bins from the solution one by one starting from the ones with witness having smaller fragility. Feasible bin means the bin capacity is not exceeded, and the bin does not contain any objects that are already packed in other bins that are copied over previously. Update the witness list and unpacked list accordingly (i.e. take out the objects that are packed from the lists);

4. Iterate through the bins that are already open, which are the ones with $y_i = 1$, and fill them up by picking the next unpacked item if possible;

5. Iterate through the witness list from the smallest witness and fill up with any unpacked objects.

6. If there are still unpacked items, open a bin and fill it up.

7. Once all items are packed, update the incumbent accordingly.

8. Repeat the above steps at every iteration of the Lagrangian algorithm when a subproblem solution is obtained.

4.2 Greedy Heuristic

This section describes a greedy heuristic that finds an upper bound for the problem. The items are presorted by fragility in non-decreasing order. The basic idea of the heuristic is to pick an item to be the witness of an empty bin, filling this bin up with other suitable items. This process is repeated until all items are packed. The key here is to first pick the right witness; and second is how to pick the items to fill an open bin.

Consider the fragility constraint (3.13). It implies that the item with smallest fragility must be a witness. Furthermore, for any subset of items yet to be packed, the one with smallest fragility must be a witness. Therefore, the witness is the item with smallest fragility picked from all items left to be packed.

Once a witness is decided, the remaining capacity is then filled. When choosing the item to be packed, two aspects are considered: first use up as much available capacity as possible; and second, pack items with smaller fragility so that the next witness has larger capacity. This selection process can be modelled as an integer program. Given the witness $\bar{i}$, we want to identify a set of x_i for $i = 1, \ldots, n$, taking value 1 if the item is picked and 0 otherwise.

$$[GH2]: \quad \max \quad \sum_{i \in U} \left(\frac{w_i}{f_i}\right) x_i \tag{4.1}$$

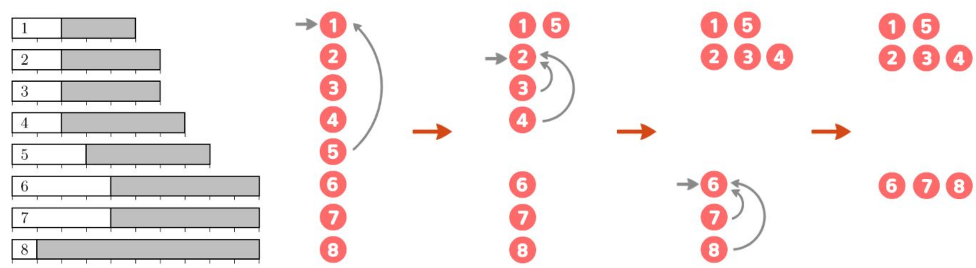
$$\text{s.t.} \quad \sum_{i \in U} w_i x_i + s = f_{\bar{i}} - w_{\bar{i}} \qquad i \in U \tag{4.2}$$

$$x_i \in \{0, 1\} \qquad i \in U \tag{4.3}$$

The objective function maximizes ratio of weight to fragility, thus the items with minimum ratio of fragility to capacity would tend to be picked. Constraint (4.2) ensures the fragility constraint is met. U is the set of items left to be packed.

An example is illustrated below. In this example, first, item 1 is chosen as witness. Based on its weight and fragility, solving $[GH2]$ gives the solution item 5 alone. Once item 5 is packed with item 1, the next item with minimum fragility is 2 (in case of ties, break the tie with the item of smaller number), and solving again gives items 3 and 4. The process is repeated until all items are packed. The optimal solution is given in the last column.

Figure 4.1: An illustration of the greedy heuristic



4.3 Tabu Search

Tabu search has been proven to be efficient in solving sequencing problems. Here, we intend to model the BPPFO as a sequence of objects, and starting from the first object as the witness, fill it up with subsequent objects in the sequence and so on. Then, the next available object will be used as the next witness. The way to find the optimal assignment of objects would essentially turn into the problem of sequencing the objects.

The scheme involves the neighbourhood of a solution, the tabu list, the fitness evaluation, and the stopping criteria.

Neighbourhood: The neighbours are defined by placing an item into the second place in sequence, and shifting everything else except for the first object, which is never moved as it is the item with smallest fragility, or, the witness.

Tabu list: The tabu list consists of solutions that were used to update the incumbent. The first solution in the list, which is the solution that was appended the earliest, is removed from the tabu list first.

Fitness evaluation: The sequence is examined and the number of bins used to pack according to such sequence is calculated.

Stopping criteria: The algorithm will stop if a lower bound is met or a maximum number of iteration is reached.

The initial solution is a sequence from 1 up to number of items. While the stopping criteria is not met, explore the neighbours of a solution and evaluate their fitness. The current best solution is updated when a better or equally good solution is found. Once the solution is updated, it is appended to the tabu list. The process halts when the stopping criterion is met.

4.4 Conclusion

Several heuristics are proposed in this section. Other than the Lagrangian heuristic, which is embedded within the Lagrangian approach, the rest are standalone.

Chapter 5

A Graph Based Model and Solution Approach

In this chapter we present a brand new formulation to the problem. We model the problem based on a graph, and solve for the minimum number of paths to find the minimum number of bins needed.

The items are represented as nodes on a graph. Two dummy nodes are added to represent an origin and a destination. The packing of items in a bin is represented by a path from the origin to the destination. The nodes on this path would be the items to be packed in one bin. The first node that connects with the origin is the witness of the bin. Details are discussed below.

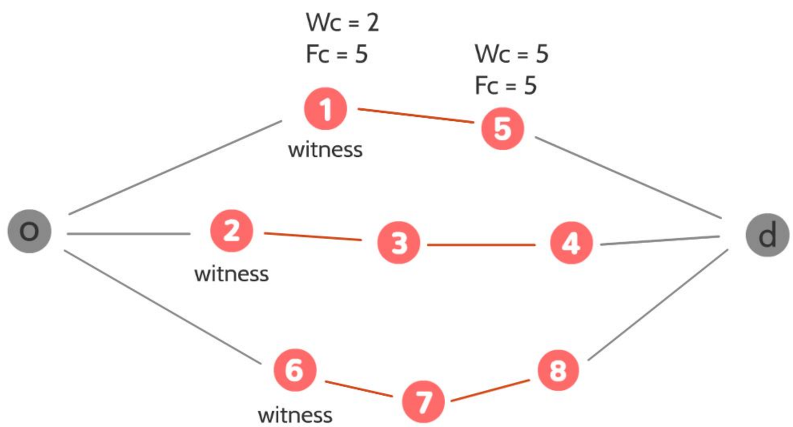


Figure 5.1: An illustration of the graph-based approach

5.1 An Illustration

In this example, there are 8 items to be packed. The optimal solution is shown in figure 5.1. The origin connects to items 1, 2, and 6, which are labelled as witnesses. The rule for the path is that the weights of the items accumulated along this path shall not exceed the fragility of the path, which is to remain the same throughout the path. The nodes after the witness also have to have greater fragility than the witness. For example, in this case, item 1 has a weight of 2 and fragility of 5, item 5 has weight 3 and fragility of 8. Item 1 is chosen to be the witness. At this point, the weight accumulated on this path is 2, and fragility is 5. Item 5 is chosen to be appended to this path, resulting in an increase of the weight accumulated to be 5. This does not violate the fragility of this path. The bin is full now. Once the bin is filled up, the path would end at the destination. Counting the number of arcs going out from the origin or going into the destination would give the number of bins used.

5.2 Formulation

The formulation for this graphical approach is presented in this section. Two new variables are introduced, accumulated weight w^c and minimum fragility f^c. w^c will accumulate item weights as it is appended to the path, whereas f^c will remain as the fragility of the witness along the entire path. x_{ij} is a binary decision variable; it takes value 1 if arc(ij) is in the solution and 0 otherwise; i and $j \in o, 1, 3...n, d$, where o is origin, d is destination, and $1, ..., n$ are the items. w_i^c and f_i^c are the accumulated weight and minimum fragility at each node respectively, for $i = 1, .., n$. Another variable K is defined as the number of arcs going out from the origin or going into the destination node.

$$
\begin{aligned}
[GM]: \quad & \min \quad K \\
& \text{s.t.} \quad \sum_{j=1}^{n} x_{oj} - K = 0 \qquad j = 1 \ldots, n \\
& \sum_{i=1}^{n} x_{id} - K = 0 \qquad i = 1, \ldots, n \\
& \sum_{i=o}^{n} x_{ij} = 1 \qquad j = 1, \ldots, n \\
& \sum_{j=1}^{d} x_{ij} = 1 \qquad i = 1, \ldots, n \\
& f_i \leq f_j - M(1 - x_{ij}) \qquad i = 1, \ldots, n, j = 1, \ldots, n \\
& w_i^c \leq f_i^c \qquad i = 1, \ldots, n \\
& w_j^c \geq w_i^c + w_j + M(x_{ij} - 1) \qquad i = 1, \ldots, n, j = 1, \ldots, n \\
& f_j^c \leq f_i^c + M(1 - x_{ij}) \qquad i = 1, \ldots, n, j = 1, \ldots, n \\
& f_j^c \leq f_j + M(1 - x_{oj}) \qquad j = 1, \ldots, n \\
& w_j^c \geq w_j + M(x_{oj} - 1) \qquad j = 1, \ldots, n \\
& \sum_{j=1}^{n} f_j x_{oj} \geq \sum_{j=1}^{n} w_j \\
& \sum_{i=o}^{n} \sum_{j=1}^{d} x_{ij} \leq n + K \qquad x_{ij} \in 0, 1 \qquad i = o, 1, \ldots, n,
\end{aligned}
$$

$K \geq 0$

The objective (5.1) minimizes the number of paths in the graph. Constraints (5.2) and (5.3) ensures that the number of arcs going out from the origin and into the destination is the number of paths. Constraints (5.4) and (5.5) limit the arcs going into and out from a node each to be 1 exactly. Constraint (5.6) makes sure arcs are going from lower fragility item to higher fragility item. Constraint (5.7) ensures that at each node, the accumulated weight does not exceed the minimum fragility. Constraint (5.8) sets the accumulated weight at each node. Constraint (5.9) sets the minimum fragility at each node. Constraint (5.10) relates the witness' fragility with the variable accumulated fragility. Constraint (5.11) relates the witness' weight with the variable accumulated weight. Constraint (5.12) ensures that witnesses selected provide enough total capacity for all items to be packed. Constraint (5.13) tightens the search space by having the total number of arcs in graph to be the sum of items and the number of paths used.

5.3 Implementation

The model is implemented in Matlab R2014a and solved with Cplex 12.6 mixed integer solver. It was found that without an initial solution to the solver to start from, the solver would not be able to find an optimal solution within one hour of computational time. Providing the greedy heuristic solution as an initial solution and adding an upper bound on K, the solver is able to provide an optimal solution very quickly, which is in most cases the initial solution provided.

Chapter 6

Numerical Results

The data instances were contributed by Clautiaux et al.[7]. The weight and fragility of items are generated according to certain correlations. The most difficult cases are recorded and are available at **http://www.or.unimore.it/resources.htm**. Five classes of uncorrelated and weakly correlated data are chosen to represent some of the difficult instances. Each class is characterized by 2 parameters, which determine how the weights and fragilities are generated:

Class 1: uncorrelated with parameters 1 and 3;

Class 2: uncorrelated with parameters 1 and 5;

Class 3: weakly correlated with parameters 1 and 5;

Class 4: weakly correlated with parameters 3 and 3;

Class 5: weakly correlated with parameters 3 and 4.

There are 135 instances in each class, consisting of 45 instances with 50 items to be packed, 45 with 100 items and 45 with 200 items. Details of the data instances can be found in Clautiaux et al. [7].

6.1 Numerical Results

Table 6.1 summarizes the numerical results. It lists the average lower bounds of each class obtained from Lagrangian relaxation of the compact model (LB_{lag1}), the new model (LB_{lag2}), and the fractional lower bound (LB_f) described by Bansal et al.[2]. The upper bounds listed are from the Lagrangian heuristic (UB_{LH}), greedy heuristic (UB_{GH}), and Tabu search (UB_{TS}). The data are presented by class (cl) 1 to 5; each is an average over 45 instances of the 50-item dataset. Larger data sets were not completed due to very long runtimes.

Table 6.1: Numeric Result of Lower and Upper Bounds

n	cl	LB_{lag1}	LB_{lag2}	LB_f	UB_{LH}	UB_{GH}	UB_{TS}
	1	13.18	13.18	12.96	14.36	13.38	13.9
	2	8.93	8.93	8.89	10	9.09	9.4
50	3	11.49	11.49	11.38	12.47	13.4	12.4
	4	11.49	11.49	11.42	12.31	13.2	12.4
	5	10.31	10.31	10.27	11.36	12.36	11.1
	1	25.64	25.64	25.38	28.44	25.89	27.76
	2	17.60	17.60	17.58	22.22	17.84	18.69
100	3	23.80	23.80	23.64	27.38	24.16	25.82
	4	23.09	23.09	22.91	26.47	23.53	25.36
	5	19.96	19.96	19.80	24.07	20.40	21.80

We can see from the table above that the two Lagrangian lower bounds are the same. This is intuitive as Lagrangian relaxation was applied on the same assignment constraint to both models. The fractional lower bound provides a lower bound with slightly less quality. However, as can be seen from the detailed tables in the Appendix, it has a runtime in the order of milliseconds for instances up to 200 items, while the Lagrangian lower bound takes up to 1000 seconds for the 100 items, and Matlab runs out of memory for 200 items. Therefore, the fractional lower bound is highly recommended to give a lower bound quickly when the quality of bound is not crucial.

It also shows that the greedy heuristic outperforms the other two upper bounding heuristics in every class for larger instances. In the next section, we will be comparing the performance of the greedy heuristic against best known benchmarks in the literature.

6.2 Lagrangian Lower Bounds

The table below shows the runtimes of the two Lagrangian lower bounds. Although they provide the same bounds, there is a major difference in the runtime. As discussed in chapter 3, the advantage of the new formulation is that it only solves one subproblem, whereas the compact model solves multiple subproblems to be solved. This significantly decreases the runtime of the algorithm.

Table 6.2: Comparison of Lagrangian Lower Bounds

n	cl	LB_{lag1}		LB_{lag2}	
		Sec	Gap	Sec	Gap
50	1	71.6	1.3%	**21.8**	1.3%
	2	289.8	1.8%	**37.9**	1.8%
	3	48.2	8.9%	**26.4**	8.9%
	4	990.2	7.6%	**29.9**	7.6%
	5	392.2	10.4%	**34.7**	10.4%
100	1	834.8	0.3%	**224.9**	0.3%
	2	1163.4	0.1%	**434.7**	0.1%
	3	828.4	0.4%	**231.1**	0.4%
	4	759.6	2.3%	**294.6**	2.3%
	5	950.0	1.2%	**329.3**	1.2%
Overall avg.		632.8	3.4%	166.5	3.4%

6.3 Performance of Greedy Heuristic Against Best Benchmarks

We compare the performance of the greedy heuristic against results from Martinez et al. [19]. The gap is calculated using a lower bound based on branch-and-price, and an upper bound based on Variable Neighbourhood Search. [7] Opt is the number of instances that are solved to optimality, and Sec is CPU time in seconds. The gaps for UB_{GH} n50 and n100 instances are calculated using lower bounds from LB_{Lag1}; whereas n200 are using lower bounds from Martinex et al. [19] branch-and-price.

We can see from the table, that even with the largest instances, the greedy heuristic still has a very competitive runtime. If we look at the number of optimal solutions reached, we find that for the larger instances, the greedy heuristic can

Table 6.3: Comparison of Upper Bound Quality

n	cl	Martinez el al. [19]			UB_{GH}		
		Gap	Opt	Sec	Gap	Opt	Sec
50	1	0.00%	**45**	58.62	1.50%	36	0.13
	2	0.00%	**45**	10.43	2.20%	36	0.10
	3	0.00%	**45**	47.67	2.09%	34	0.15
	4	0.00%	**45**	45.25	2.08%	34	0.16
	5	0.00%	**45**	17.18	0.85%	39	0.17
Avg/tot		0.00%	**225**	35.83	1.74%	179	0.14
100	1	0.72%	**37**	927.53	2.88%	36	0.45
	2	0.36%	**42**	306.83	4.17%	34	0.35
	3	0.29%	**42**	418.63	3.42%	29	0.53
	4	0.22%	**43**	385.22	3.98%	25	0.65
	5	0.55%	**40**	621.10	4.34%	30	0.47
Avg/tot		0.43%	**204**	531.86	3.76%	154	0.49
200	1	1.03%	22	1922.27	1.29%	**23**	1.91
	2	1.37%	**24**	1712.02	0.99%	**24**	2.54
	3	0.87%	**27**	1587.09	1.59%	16	3.27
	4	1.24%	**20**	2134.94	1.46%	17	3.25
	5	1.12%	**25**	1806.98	0.95%	**25**	2.84
Avg/tot		1.13%	**118**	1832.66	1.26%	105	2.76
Overall		0.52%	**547**	800.12	2.25%	438	1.13

provide some equally good or even better solutions than Martinez et al. [19].

Chapter 7

Conclusion

We explored various methods to solve the bin packing problem with fragile objects. We used a Lagrangian approach to find a lower bound using both a compact model presented by Clautiaux et al.[7] and a new formulation. While the two lower bounds obtained are exactly the same, the new formulation requires significantly less runtime as it only needs to solve one subproblem. We then attempted several methods to obtain a feasible solution and upper bound for the problem. It was found that the greedy heuristic outperforms other heuristics not only in terms of quality of the bound, which is closer to optimal, but also in terms of runtime. It is also comparable to the best benchmark in the literature. We found that the greedy heuristic performs especially well for larger data instances. For most of the time, it provides a reasonable bound within very short runtime. Lastly, we presented a whole new solution approach to the problem. We modeled the problem using a graph where finding paths on the graph lead to optimal packing of the bins. The direct solution of the model was not able to give a feasible solution within reasonable time, except when an initial solution is given. Therefore, further research can be devoted to improving the solution of the graph-based model.

Appendix A

Numerical Results

Table A.1: Legends for Tables

n:	number of items in instance
cl:	class of instance
VNS:	Variable Neighbour Search [7]
CM:	compact model [7]
UB_0, UB:	upper bound using VNS [7]
LB:	lower bound from solving CM [7]
LB_{lag1}:	Lagrangian lower bound of [P1]
LB_{lag2}:	Lagrangian lower bound of [P2]
LB_f:	fractional lower bound
UB_{LH}:	Lagrangian heuristic upper bound
UB_{GH}:	greedy heuristic upper bound
UB_{TS}:	tabu search upper bound
Sec:	CPU time in second
Gap:	gap of bounds
Opt:	number of instances with optimality reached

			VNS+CM							Lagrangian+GH					
Instance	**n**	**cl**	**UB_0**	**L$_f$**	**LB**	**UB**	**gap**	**sec**	**opt**	**LB_lag1**	**sec**	**LB_lag2**	**GH_UB**	**sec**	**opt**
N1C1W1_CL1_1_3_A	50	1	13	13	13	13	0.00%	0.00	1	13	81.117	13	13	0.61	1
N1C1W1_CL1_1_3_B	50	1	15	14	15	15	0.00%	151.32	1	15	83.296	15	15	0.18	1
N1C1W1_CL1_1_3_C	50	1	11	11	11	11	0.00%	0.00	1	11	95.578	11	11	0.10	1
N1C1W1_CL1_1_3_D	50	1	14	13	13	13	0.00%	154.66	1	13	82.043	13	14	0.12	0
N1C1W1_CL1_1_3_E	50	1	13	13	13	13	0.00%	0.01	1	13	71.851	13	13	0.18	1
N1C1W2_CL1_1_3_A	50	1	14	13	14	14	0.00%	151.32	1	14	77.186	14	14	0.20	1
N1C1W2_CL1_1_3_B	50	1	15	14	15	15	0.00%	151.40	1	15	58.985	15	15	0.12	1
N1C1W2_CL1_1_3_C	50	1	16	16	16	16	0.00%	0.00	1	16	47.49	16	16	0.14	1
N1C1W2_CL1_1_3_D	50	1	15	15	15	15	0.00%	0.00	1	15	63.203	15	15	0.09	1
N1C1W2_CL1_1_3_E	50	1	17	16	17	17	0.00%	151.26	1	17	54.704	17	17	0.10	1
N1C1W4_CL1_1_3_A	50	1	16	16	16	16	0.00%	0.00	1	16	27.264	16	16	0.13	1
N1C1W4_CL1_1_3_B	50	1	18	17	18	18	0.00%	152.51	1	18	28.532	18	18	0.11	1
N1C1W4_CL1_1_3_C	50	1	16	16	16	16	0.00%	0.00	1	16	34.987	16	16	0.11	1
N1C1W4_CL1_1_3_D	50	1	17	17	17	17	0.00%	74.21	1	17	31.991	17	18	0.16	0
N1C1W4_CL1_1_3_E	50	1	17	17	17	17	0.00%	0.63	1	17	35.351	17	17	0.10	1
N1C2W1_CL1_1_3_A	50	1	11	11	11	11	0.00%	0.000	1	11	76.263	11	11	0.06	1
N1C2W1_CL1_1_3_B	50	1	13	12	13	13	0.00%	150.93	1	13	65.678	13	13	0.12	1
N1C2W1_CL1_1_3_C	50	1	12	12	12	12	0.00%	0.00	1	12	59.985	12	12	0.09	1
N1C2W1_CL1_1_3_D	50	1	12	11	12	12	0.00%	328.73	1	11	74.339	11	12	0.09	0
N1C2W1_CL1_1_3_E	50	1	10	9	10	10	0.00%	150.89	1	10	86.048	10	10	0.08	1
N1C2W2_CL1_1_3_A	50	1	13	13	13	13	0.00%	0.00	1	13	70.713	13	13	0.13	1
N1C2W2_CL1_1_3_B	50	1	14	14	14	14	0.00%	0.00	1	14	56.347	14	14	0.14	1
N1C2W2_CL1_1_3_C	50	1	14	14	14	14	0.00%	0.01	1	14	72.639	14	14	0.10	1
N1C2W2_CL1_1_3_D	50	1	13	13	13	13	0.00%	0.00	1	13	59.714	13	13	0.17	1
N1C2W2_CL1_1_3_E	50	1	16	15	15	15	0.00%	178.47	1	15	48.347	15	16	0.13	0
N1C2W4_CL1_1_3_A	50	1	15	14	15	15	0.00%	151.00	1	15	46.58	15	15	0.14	1
N1C2W4_CL1_1_3_B	50	1	15	15	15	15	0.00%	0.00	1	15	42.138	15	15	0.09	1
N1C2W4_CL1_1_3_C	50	1	15	14	15	15	0.00%	151.89	1	15	58.706	15	15	0.10	1
N1C2W4_CL1_1_3_D	50	1	14	14	14	14	0.00%	0.00	1	14	52.285	14	14	0.11	1
N1C2W4_CL1_1_3_E	50	1	15	15	15	15	0.00%	0.00	1	15	67.877	15	15	0.13	1
N1C3W1_CL1_1_3_A	50	1	10	9	9	9	0.00%	155.47	1	9	77.521	9	10	0.08	0
N1C3W1_CL1_1_3_B	50	1	9	9	9	9	0.00%	0.00	1	9	79.007	9	9	0.08	1
N1C3W1_CL1_1_3_C	50	1	10	10	10	10	0.00%	0.00	1	10	83.158	10	10	0.11	1
N1C3W1_CL1_1_3_D	50	1	11	11	11	11	0.00%	0.00	1	11	67.639	11	11	0.09	1
N1C3W1_CL1_1_3_E	50	1	9	9	9	9	0.00%	0.00	1	9	65.155	9	9	0.07	1
N1C3W2_CL1_1_3_A	50	1	11	11	11	11	0.00%	0.00	1	11	80.39	11	11	0.10	1
N1C3W2_CL1_1_3_B	50	1	12	11	12	12	0.00%	197.72	1	11	61.934	11	12	0.17	0
N1C3W2_CL1_1_3_C	50	1	12	12	12	12	0.00%	0.00	1	12	91.939	12	13	0.12	0
N1C3W2_CL1_1_3_D	50	1	12	11	12	12	0.00%	840.32	1	11	71.238	11	12	0.17	0
N1C3W2_CL1_1_3_E	50	1	12	12	12	12	0.00%	0.00	1	12	78.394	12	12	0.12	1
N1C3W4_CL1_1_3_A	50	1	12	12	12	12	0.00%	0.00	1	12	63.823	12	12	0.14	1
N1C3W4_CL1_1_3_B	50	1	13	12	12	12	0.00%	339.23	1	12	76.338	12	13	0.17	0
N1C3W4_CL1_1_3_C	50	1	13	13	13	13	0.00%	0.00	1	13	61.436	13	13	0.10	1
N1C3W4_CL1_1_3_D	50	1	12	12	12	12	0.00%	0.00	1	12	68.998	12	12	0.11	1
N1C3W4_CL1_1_3_E	50	1	13	12	13	13	0.00%	151.64	1	13	68.307	13	13	0.12	1

Table A.2: Detailed Comparison for N50 Class1

			VNS+CM							Lagrangian+GH					
Instance	**n**	**cl**	**UB_0**	**L$_f$**	**LB**	**UB**	**gap**	**sec**	**opt**	**LB_lag1**	**sec**	**LB_lag2**	**GH_UB**	**sec**	**opt**
N1C1W1_CL1_1_5_A	50	2	9	9	9	9	0.00%	0.00	1	9	86.706	9	9	0.15364	1
N1C1W1_CL1_1_5_B	50	2	10	10	10	10	0.00%	0.00	1	10	68.779	10	10	0.075845	1
N1C1W1_CL1_1_5_C	50	2	8	7	7	7	0.00%	153.65	1	7	96.488	7	8	0.084575	0
N1C1W1_CL1_1_5_D	50	2	9	9	9	9	0.00%	0.00	1	9	78.772	9	9	0.06078	1
N1C1W1_CL1_1_5_E	50	2	9	9	9	9	0.00%	0.00	1	9	91.728	9	9	0.089802	1
N1C1W2_CL1_1_5_A	50	2	9	9	9	9	0.00%	1.03	1	9	79.811	9	10	0.14726	0
N1C1W2_CL1_1_5_B	50	2	10	10	10	10	0.00%	0.00	1	10	97.282	10	11	0.097793	0
N1C1W2_CL1_1_5_C	50	2	11	11	11	11	0.00%	0.00	1	11	69.646	11	11	0.15181	1
N1C1W2_CL1_1_5_D	50	2	10	10	10	10	0.00%	0.00	1	10	57.323	10	10	0.14619	1
N1C1W2_CL1_1_5_E	50	2	11	11	11	11	0.00%	2.23	1	11	75.23	11	12	0.13433	0
N1C1W4_CL1_1_5_A	50	2	11	11	11	11	0.00%	0.00	1	11	61.772	11	11	0.14517	1
N1C1W4_CL1_1_5_B	50	2	12	12	12	12	0.00%	0.00	1	12	79.154	12	12	0.10341	1
N1C1W4_CL1_1_5_C	50	2	11	11	11	11	0.00%	0.00	1	11	70.199	11	11	0.14096	1
N1C1W4_CL1_1_5_D	50	2	12	12	12	12	0.00%	0.00	1	12	75.253	12	12	0.093367	1
N1C1W4_CL1_1_5_E	50	2	12	11	12	12	0.00%	190.33	1	11	81.8	11	12	0.13603	0
N1C2W1_CL1_1_5_A	50	2	7	7	7	7	0.00%	0.00	1	7	86.356	7	8	0.071684	0
N1C2W1_CL1_1_5_B	50	2	9	9	9	9	0.00%	0.00	1	9	98.023	9	9	0.061071	1
N1C2W1_CL1_1_5_C	50	2	8	8	8	8	0.00%	0.00	1	8	85.521	8	8	0.099499	1
N1C2W1_CL1_1_5_D	50	2	8	8	8	8	0.00%	0.00	1	8	88.398	8	8	0.064112	1
N1C2W1_CL1_1_5_E	50	2	7	7	7	7	0.00%	0.00	1	7	102.94	7	7	0.061922	1
N1C2W2_CL1_1_5_A	50	2	9	9	9	9	0.00%	0.00	1	9	111.01	9	9	0.068972	1
N1C2W2_CL1_1_5_B	50	2	10	9	10	10	0.00%	163.12	1	9	84.833	9	10	0.15629	0
N1C2W2_CL1_1_5_C	50	2	10	10	10	10	0.00%	0.00	1	10	93.194	10	10	0.1758	1
N1C2W2_CL1_1_5_D	50	2	9	9	9	9	0.00%	0.00	1	9	97.924	9	9	0.11452	1
N1C2W2_CL1_1_5_E	50	2	10	10	10	10	0.00%	0.00	1	10	71.801	10	10	0.1052	1
N1C2W4_CL1_1_5_A	50	2	10	10	10	10	0.00%	0.00	1	10	88.869	10	10	0.15419	1
N1C2W4_CL1_1_5_B	50	2	10	10	10	10	0.00%	0.00	1	10	71.524	10	10	0.10048	1
N1C2W4_CL1_1_5_C	50	2	10	10	10	10	0.00%	0.00	1	10	71.488	10	10	0.086501	1
N1C2W4_CL1_1_5_D	50	2	9	9	9	9	0.00%	1.07	1	9	87.694	9	9	0.086362	1
N1C2W4_CL1_1_5_E	50	2	10	10	10	10	0.00%	0.00	1	10	70.866	10	10	0.099021	1
N1C3W1_CL1_1_5_A	50	2	6	6	6	6	0.00%	0.00	1	6	103.17	6	7	0.067655	0
N1C3W1_CL1_1_5_B	50	2	6	6	6	6	0.00%	0.00	1	6	113.29	6	6	0.044071	1
N1C3W1_CL1_1_5_C	50	2	7	7	7	7	0.00%	0.00	1	7	114.69	7	7	0.043728	1
N1C3W1_CL1_1_5_D	50	2	7	7	7	7	0.00%	0.00	1	7	95.354	7	7	0.046743	1
N1C3W1_CL1_1_5_E	50	2	6	6	6	6	0.00%	0.00	1	6	120.64	6	6	0.051104	1
N1C3W2_CL1_1_5_A	50	2	7	7	7	7	0.00%	0.00	1	7	109.54	7	7	0.10222	1
N1C3W2_CL1_1_5_B	50	2	8	8	8	8	0.00%	0.00	1	8	98.503	8	8	0.12408	1
N1C3W2_CL1_1_5_C	50	2	8	8	8	8	0.00%	0.00	1	8	113.57	8	8	0.077544	1
N1C3W2_CL1_1_5_D	50	2	8	8	8	8	0.00%	0.00	1	8	120.36	8	8	0.089491	1
N1C3W2_CL1_1_5_E	50	2	8	8	8	8	0.00%	0.00	1	8	97.723	8	8	0.13175	1
N1C3W4_CL1_1_5_A	50	2	8	8	8	8	0.00%	0.00	1	8	107.23	8	8	0.15021	1
N1C3W4_CL1_1_5_B	50	2	8	8	8	8	0.00%	0.00	1	8	92.238	8	9	0.12318	0
N1C3W4_CL1_1_5_C	50	2	9	9	9	9	0.00%	0.00	1	9	80.721	9	9	0.097527	1
N1C3W4_CL1_1_5_D	50	2	8	8	8	8	0.00%	0.00	1	8	106.91	8	8	0.096046	1
N1C3W4_CL1_1_5_E	50	2	9	9	9	9	0.00%	0.00	1	9	115.23	9	9	0.089157	1

Table A.3: Detailed Comparison for N50 Class2

			VNS+CM							Lagrangian+GH					
Instance	**n**	**cl**	**UB_0**	**L$_f$**	**LB**	**UB**	**gap**	**sec**	**opt**	**LB_lag1**	**sec**	**LB_lag2**	**GH_UB**	**sec**	**opt**
N1C1W1_CL2_1_5_A	50	3	11	11	11	11	0.00%	0.00	1	11	48.403	11	11	0.099985	1
N1C1W1_CL2_1_5_B	50	3	13	13	13	13	0.00%	0.00	1	13	40.265	13	13	0.13769	1
N1C1W1_CL2_1_5_C	50	3	10	10	10	10	0.00%	0.00	1	10	61.862	10	10	0.086198	1
N1C1W1_CL2_1_5_D	50	3	12	12	12	12	0.00%	0.00	1	12	54.099	12	12	0.087282	1
N1C1W1_CL2_1_5_E	50	3	11	11	11	11	0.00%	0.00	1	11	58.242	11	11	0.13987	1
N1C1W2_CL2_1_5_A	50	3	12	12	12	12	0.00%	0.00	1	12	56.152	12	12	0.20566	1
N1C1W2_CL2_1_5_B	50	3	13	12	13	13	0.00%	153.12	1	13	55.59	13	13	0.19064	1
N1C1W2_CL2_1_5_C	50	3	14	13	14	14	0.00%	296.03	1	13	49.785	13	14	0.1934	0
N1C1W2_CL2_1_5_D	50	3	13	13	13	13	0.00%	0.00	1	13	69.096	13	13	0.17026	1
N1C1W2_CL2_1_5_E	50	3	14	14	14	14	0.00%	0.00	1	14	53.194	14	14	0.18712	1
N1C1W4_CL2_1_5_A	50	3	14	13	14	14	0.00%	170.01	1	13	68.566	13	14	0.23467	0
N1C1W4_CL2_1_5_B	50	3	14	14	14	14	0.00%	0.01	1	14	61.682	14	14	0.17574	1
N1C1W4_CL2_1_5_C	50	3	14	13	13	13	0.00%	237.17	1	13	41.703	13	14	0.1697	0
N1C1W4_CL2_1_5_D	50	3	14	14	14	14	0.00%	0.20	1	14	43.343	14	15	0.13946	0
N1C1W4_CL2_1_5_E	50	3	14	14	14	14	0.00%	0.02	1	14	58.066	14	14	0.17723	1
N1C2W1_CL2_1_5_A	50	3	10	10	10	10	0.00%	0.00	1	10	63.594	10	10	0.16187	1
N1C2W1_CL2_1_5_B	50	3	11	11	11	11	0.00%	0.00	1	11	70.708	11	11	0.18706	1
N1C2W1_CL2_1_5_C	50	3	11	11	11	11	0.00%	0.00	1	11	58.647	11	11	0.10302	1
N1C2W1_CL2_1_5_D	50	3	11	10	10	10	0.00%	263.81	1	10	62.72	10	11	0.083768	0
N1C2W1_CL2_1_5_E	50	3	9	9	9	9	0.00%	0.00	1	9	65.805	9	9	0.10402	1
N1C2W2_CL2_1_5_A	50	3	11	11	11	11	0.00%	0.05	1	11	64.035	11	12	0.17654	0
N1C2W2_CL2_1_5_B	50	3	12	12	12	12	0.00%	0.13	1	12	54.993	12	12	0.14689	1
N1C2W2_CL2_1_5_C	50	3	13	12	12	12	0.00%	1312.72	1	12	57.113	12	12	0.13841	1
N1C2W2_CL2_1_5_D	50	3	11	11	11	11	0.00%	0.00	1	11	70.837	11	11	0.19455	1
N1C2W2_CL2_1_5_E	50	3	13	13	13	13	0.00%	0.00	1	13	60.907	13	13	0.14325	1
N1C2W4_CL2_1_5_A	50	3	13	12	12	13	7.69%	3600.00	0	12	70.905	12	13	0.24598	0
N1C2W4_CL2_1_5_B	50	3	13	12	13	13	0.00%	153.80	1	13	68.143	13	13	0.21006	1
N1C2W4_CL2_1_5_C	50	3	13	12	13	13	0.00%	228.52	1	12	68.704	12	13	0.13469	0
N1C2W4_CL2_1_5_D	50	3	12	12	12	12	0.00%	0.00	1	12	52.848	12	12	0.18083	1
N1C2W4_CL2_1_5_E	50	3	13	12	13	13	0.00%	152.33	1	13	60.83	13	13	0.13571	1
N1C3W1_CL2_1_5_A	50	3	9	9	9	9	0.00%	0.00	1	9	82.2	9	9	0.07406	1
N1C3W1_CL2_1_5_B	50	3	9	9	9	9	0.00%	0.00	1	9	72.758	9	9	0.10053	1
N1C3W1_CL2_1_5_C	50	3	10	9	10	10	0.00%	175.38	1	9	80.563	9	10	0.1247	0
N1C3W1_CL2_1_5_D	50	3	10	10	10	10	0.00%	0.00	1	10	78.874	10	10	0.098622	1
N1C3W1_CL2_1_5_E	50	3	9	9	9	9	0.00%	0.00	1	9	69.621	9	9	0.080453	1
N1C3W2_CL2_1_5_A	50	3	10	10	10	10	0.00%	0.00	1	10	80.053	10	10	0.094466	1
N1C3W2_CL2_1_5_B	50	3	11	10	11	11	0.00%	883.49	1	10	84.957	10	11	0.19348	0
N1C3W2_CL2_1_5_C	50	3	11	11	11	11	0.00%	0.00	1	11	69.636	11	11	0.15613	1
N1C3W2_CL2_1_5_D	50	3	11	10	11	11	0.00%	310.10	1	10	68.055	10	11	0.1849	0
N1C3W2_CL2_1_5_E	50	3	11	11	11	11	0.00%	0.00	1	11	66.318	11	11	0.13456	1
N1C3W4_CL2_1_5_A	50	3	11	11	11	11	0.00%	0.00	1	11	61.667	11	11	0.15999	1
N1C3W4_CL2_1_5_B	50	3	11	11	11	11	0.00%	0.00	1	11	66.95	11	11	0.1156	1
N1C3W4_CL2_1_5_C	50	3	12	11	12	12	0.00%	152.51	1	12	68.195	12	12	0.18134	1
N1C3W4_CL2_1_5_D	50	3	11	11	11	11	0.00%	0.00	1	11	69.164	11	11	0.16164	1
N1C3W4_CL2_1_5_E	50	3	11	11	11	11	0.00%	36.03	1	11	54.966	11	11	0.14832	1

Table A.4: Detailed Comparison for N50 Class3

			VNS+CM							Lagrangian+GH					
Instance	n	cl	UB_0	L_f	LB	UB	gap	sec	opt	LB_lag1	sec	LB_lag2	GH_UB	sec	opt
N1C1W1_CL2_3_3_A	50	4	11	11	11	11	0.00%	0.78	1	11	73.744	11	12	0.09159	0
N1C1W1_CL2_3_3_B	50	4	13	13	13	13	0.00%	0.00	1	13	60.88	13	13	0.14673	1
N1C1W1_CL2_3_3_C	50	4	10	10	10	10	0.00%	0.00	1	10	61.878	10	10	0.12815	1
N1C1W1_CL2_3_3_D	50	4	12	12	12	12	0.00%	6.80	1	12	46.095	12	12	0.12571	1
N1C1W1_CL2_3_3_E	50	4	12	11	12	12	0.00%	157.09	1	12	53.402	12	12	0.12666	1
N1C1W2_CL2_3_3_A	50	4	12	12	12	12	0.00%	0.00	1	12	59.047	12	12	0.16344	1
N1C1W2_CL2_3_3_B	50	4	13	12	13	13	0.00%	151.93	1	13	70.066	13	13	0.18476	1
N1C1W2_CL2_3_3_C	50	4	14	14	14	14	0.00%	0.00	1	14	57.3	14	14	0.15418	1
N1C1W2_CL2_3_3_D	50	4	13	13	13	13	0.00%	0.00	1	13	58.627	13	13	0.18944	1
N1C1W2_CL2_3_3_E	50	4	14	13	14	14	0.00%	164.78	1	14	54.132	14	14	0.19659	1
N1C1W4_CL2_3_3_A	50	4	13	13	13	13	0.00%	0.00	1	13	66.376	13	13	0.18202	1
N1C1W4_CL2_3_3_B	50	4	14	14	14	14	0.00%	0.00	1	14	50.753	14	14	0.16001	1
N1C1W4_CL2_3_3_C	50	4	13	13	13	13	0.00%	0.02	1	13	40.957	13	13	0.17203	1
N1C1W4_CL2_3_3_D	50	4	15	14	15	15	0.00%	168.45	1	15	40.855	15	15	0.18474	1
N1C1W4_CL2_3_3_E	50	4	15	14	14	14	0.00%	199.80	1	14	31.026	14	15	0.15492	0
N1C2W1_CL2_3_3_A	50	4	10	10	10	10	0.00%	0.00	1	10	65.694	10	10	0.11462	1
N1C2W1_CL2_3_3_B	50	4	11	11	11	11	0.00%	0.00	1	11	81.85	11	11	0.12053	1
N1C2W1_CL2_3_3_C	50	4	11	11	11	11	0.00%	0.00	1	11	68.942	11	11	0.1116	1
N1C2W1_CL2_3_3_D	50	4	11	11	11	11	0.00%	0.00	1	11	73.328	11	11	0.099537	1
N1C2W1_CL2_3_3_E	50	4	9	9	9	9	0.00%	0.00	1	9	76.778	9	9	0.10927	1
N1C2W2_CL2_3_3_A	50	4	12	11	12	12	0.00%	367.37	1	11	61.485	11	12	0.17622	0
N1C2W2_CL2_3_3_B	50	4	12	12	12	12	0.00%	0.00	1	12	83.921	12	12	0.19491	1
N1C2W2_CL2_3_3_C	50	4	12	12	12	12	0.00%	0.04	1	12	73.608	12	12	0.16745	1
N1C2W2_CL2_3_3_D	50	4	12	11	12	12	0.00%	2349.58	1	11	71.47	11	12	0.18723	0
N1C2W2_CL2_3_3_E	50	4	13	13	13	13	0.00%	0.00	1	13	67.442	13	13	0.13811	1
N1C2W4_CL2_3_3_A	50	4	12	12	12	12	0.00%	0.00	1	12	62.864	12	12	0.16761	1
N1C2W4_CL2_3_3_B	50	4	13	12	13	13	0.00%	170.20	1	12	58.272	12	13	0.24841	0
N1C2W4_CL2_3_3_C	50	4	12	12	12	12	0.00%	0.05	1	12	53.268	12	12	0.14496	1
N1C2W4_CL2_3_3_D	50	4	12	12	12	12	0.00%	0.00	1	12	54.056	12	12	0.16253	1
N1C2W4_CL2_3_3_E	50	4	13	12	13	13	0.00%	373.25	1	12	69.81	12	13	0.16888	0
N1C3W1_CL2_3_3_A	50	4	9	9	9	9	0.00%	0.00	1	9	74.088	9	9	0.061122	1
N1C3W1_CL2_3_3_B	50	4	10	9	9	9	0.00%	489.48	1	9	91.594	9	10	0.126	0
N1C3W1_CL2_3_3_C	50	4	10	10	10	10	0.00%	0.00	1	9	71.528	9	10	0.10499	0
N1C3W1_CL2_3_3_D	50	4	10	10	10	10	0.00%	0.00	1	10	77.773	10	10	0.19521	1
N1C3W1_CL2_3_3_E	50	4	9	9	9	9	0.00%	0.00	1	9	69.148	9	9	0.10114	1
N1C3W2_CL2_3_3_A	50	4	10	10	10	10	0.00%	0.00	1	10	79.75	10	10	0.22041	1
N1C3W2_CL2_3_3_B	50	4	11	10	11	11	0.00%	295.86	1	10	61.27	10	11	0.21246	0
N1C3W2_CL2_3_3_C	50	4	11	11	11	11	0.00%	0.00	1	11	67.421	11	11	0.1975	1
N1C3W2_CL2_3_3_D	50	4	11	10	10	10	0.00%	261.50	1	10	75.654	10	11	0.12608	0
N1C3W2_CL2_3_3_E	50	4	11	11	11	11	0.00%	0.00	1	11	78.673	11	11	0.26207	1
N1C3W4_CL2_3_3_A	50	4	11	11	11	11	0.00%	0.00	1	11	73.572	11	11	0.205	1
N1C3W4_CL2_3_3_B	50	4	11	11	11	11	0.00%	0.00	1	11	68.253	11	11	0.28737	1
N1C3W4_CL2_3_3_C	50	4	12	11	12	12	0.00%	227.90	1	11	69.401	11	12	0.22098	0
N1C3W4_CL2_3_3_D	50	4	11	11	11	11	0.00%	0.00	1	11	67.89	11	11	0.13728	1
N1C3W4_CL2_3_3_E	50	4	11	11	11	11	0.00%	0.02	1	11	82.408	11	11	0.25084	1

Table A.5: Detailed Comparison for N50 Class4

			VNS+CM							Lagrangian+GH					
Instance	n	cl	UB_0	L_f	LB	UB	gap	sec	opt	LB_lag1	sec	LB_lag2	GH_UB	sec	opt
N1C1W1_CL2_3_4_A	50	5	10	10	10	10	0.00%	0.00	1	10	69.637	10	10	0.15322	1
N1C1W1_CL2_3_4_B	50	5	11	11	11	11	0.00%	0.00	1	11	54.884	11	12	0.25029	0
N1C1W1_CL2_3_4_C	50	5	9	9	9	9	0.00%	0.00	1	9	57.925	9	9	0.094614	1
N1C1W1_CL2_3_4_D	50	5	11	11	11	11	0.00%	0.00	1	11	62.603	11	11	0.18892	1
N1C1W1_CL2_3_4_E	50	5	10	10	10	10	0.00%	0.00	1	10	69.173	10	11	0.071784	0
N1C1W2_CL2_3_4_A	50	5	11	11	11	11	0.00%	0.00	1	11	62.451	11	11	0.15586	1
N1C1W2_CL2_3_4_B	50	5	11	11	11	11	0.00%	0.00	1	11	54.276	11	11	0.21215	1
N1C1W2_CL2_3_4_C	50	5	12	12	12	12	0.00%	0.00	1	12	62.796	12	12	0.27061	1
N1C1W2_CL2_3_4_D	50	5	11	11	11	11	0.00%	2.63	1	11	69.782	11	12	0.176	0
N1C1W2_CL2_3_4_E	50	5	12	12	12	12	0.00%	0.00	1	12	71.783	12	12	0.22292	1
N1C1W4_CL2_3_4_A	50	5	12	12	12	12	0.00%	0.00	1	12	62.494	12	12	0.16696	1
N1C1W4_CL2_3_4_B	50	5	13	12	13	13	0.00%	306.54	1	12	63.227	12	13	0.2685	0
N1C1W4_CL2_3_4_C	50	5	12	12	12	12	0.00%	0.00	1	12	58.297	12	12	0.28741	1
N1C1W4_CL2_3_4_D	50	5	13	12	13	13	0.00%	151.51	1	13	70.552	13	13	0.19124	1
N1C1W4_CL2_3_4_E	50	5	13	12	13	13	0.00%	169.63	1	13	55.722	13	13	0.21684	1
N1C2W1_CL2_3_4_A	50	5	9	9	9	9	0.00%	0.00	1	9	59.384	9	9	0.06567	1
N1C2W1_CL2_3_4_B	50	5	10	10	10	10	0.00%	0.00	1	10	72.871	10	10	0.11056	1
N1C2W1_CL2_3_4_C	50	5	10	10	10	10	0.00%	0.00	1	10	82.129	10	10	0.12168	1
N1C2W1_CL2_3_4_D	50	5	10	10	10	10	0.00%	0.00	1	10	74.855	10	10	0.11515	1
N1C2W1_CL2_3_4_E	50	5	8	8	8	8	0.00%	0.00	1	8	88.176	8	8	0.058668	1
N1C2W2_CL2_3_4_A	50	5	10	10	10	10	0.00%	0.05	1	10	76.081	10	10	0.21366	1
N1C2W2_CL2_3_4_B	50	5	11	11	11	11	0.00%	0.00	1	11	68.653	11	11	0.26803	1
N1C2W2_CL2_3_4_C	50	5	11	11	11	11	0.00%	0.00	1	11	61.167	11	11	0.19565	1
N1C2W2_CL2_3_4_D	50	5	10	10	10	10	0.00%	0.00	1	10	54.255	10	10	0.16674	1
N1C2W2_CL2_3_4_E	50	5	12	11	11	12	8.33%	3600.01	0	11	68.33	11	12	0.24112	0
N1C2W4_CL2_3_4_A	50	5	11	11	11	11	0.00%	0.00	1	11	72.67	11	11	0.17049	1
N1C2W4_CL2_3_4_B	50	5	11	11	11	11	0.00%	0.00	1	11	61.469	11	11	0.11493	1
N1C2W4_CL2_3_4_C	50	5	11	11	11	11	0.00%	0.00	1	11	82.054	11	11	0.19988	1
N1C2W4_CL2_3_4_D	50	5	11	11	11	11	0.00%	0.00	1	11	60.378	11	11	0.18207	1
N1C2W4_CL2_3_4_E	50	5	11	11	11	11	0.00%	0.00	1	11	58.155	11	11	0.21061	1
N1C3W1_CL2_3_4_A	50	5	8	8	8	8	0.00%	0.00	1	8	82.253	8	8	0.067842	1
N1C3W1_CL2_3_4_B	50	5	8	8	8	8	0.00%	0.00	1	8	89.049	8	8	0.10617	1
N1C3W1_CL2_3_4_C	50	5	9	9	9	9	0.00%	0.00	1	9	70.261	9	9	0.094778	1
N1C3W1_CL2_3_4_D	50	5	9	9	9	9	0.00%	0.00	1	9	67.013	9	9	0.079519	1
N1C3W1_CL2_3_4_E	50	5	8	8	8	8	0.00%	0.00	1	8	97.607	8	8	0.078735	1
N1C3W2_CL2_3_4_A	50	5	9	9	9	9	0.00%	0.00	1	9	63.846	9	9	0.20361	1
N1C3W2_CL2_3_4_B	50	5	9	9	9	9	0.00%	0.17	1	9	83.714	9	9	0.13024	1
N1C3W2_CL2_3_4_C	50	5	10	10	10	10	0.00%	0.00	1	10	86.853	10	10	0.39226	1
N1C3W2_CL2_3_4_D	50	5	9	9	9	9	0.00%	0.00	1	9	81.187	9	10	0.12606	0
N1C3W2_CL2_3_4_E	50	5	10	10	10	10	0.00%	0.00	1	10	78.28	10	10	0.209	1
N1C3W4_CL2_3_4_A	50	5	10	10	10	10	0.00%	0.00	1	10	80.928	10	10	0.19122	1
N1C3W4_CL2_3_4_B	50	5	10	10	10	10	0.00%	0.00	1	10	61.879	10	10	0.16577	1
N1C3W4_CL2_3_4_C	50	5	10	10	10	10	0.00%	0.01	1	10	86.031	10	10	0.19929	1
N1C3W4_CL2_3_4_D	50	5	10	10	10	10	0.00%	0.00	1	10	103.69	10	10	0.15625	1
N1C3W4_CL2_3_4_E	50	5	10	10	10	10	0.00%	0.00	1	10	172.35	10	10	0.23136	1

Table A.6: Detailed Comparison for N50 Class5

			VNS+CM					Lagrangian+GH					
Instance	**n**	**cl**	**UB**	**LB**	**gap**	**sec**	**opt**	**LB_lag1**	**sec**	**LB_lag2**	**sec**	**GH_UB**	**sec**	**opt**
N2C1W1_CL1_1_3_A	100	1	24	23	4.17%	3600.03	0	23	917.26	23	339.12	24	0.5834	0
N2C1W1_CL1_1_3_B	100	1	24	24	0.00%	0.00	1	24	895.28	24	298.31	24	0.59157	1
N2C1W1_CL1_1_3_C	100	1	23	23	0.00%	151.31	1	23	927.59	23	266.32	23	0.36614	1
N2C1W1_CL1_1_3_D	100	1	24	24	0.00%	0.00	1	24	918.48	24	262.14	24	0.3696	1
N2C1W1_CL1_1_3_E	100	1	27	27	0.00%	0.08	1	27	872.5	27	164.34	27	0.46221	1
N2C1W2_CL1_1_3_A	100	1	29	29	0.00%	150.69	1	29	784.1	29	167.83	29	0.67912	1
N2C1W2_CL1_1_3_B	100	1	29	28	3.45%	3600.05	0	28	778.06	28	179.34	28	0.43268	1
N2C1W2_CL1_1_3_C	100	1	30	30	0.00%	2193.24	1	29	696.57	29	161.84	30	0.50626	0
N2C1W2_CL1_1_3_D	100	1	33	33	0.00%	151.86	1	33	628.78	33	116.91	33	0.41586	1
N2C1W2_CL1_1_3_E	100	1	30	30	0.00%	151.32	1	30	680.69	30	153.58	30	0.52185	1
N2C1W4_CL1_1_3_A	100	1	33	33	0.00%	160.91	1	33	619.98	33	110.22	33	0.45371	1
N2C1W4_CL1_1_3_B	100	1	32	32	0.00%	446.53	1	32	660.04	32	138.54	33	0.45819	0
N2C1W4_CL1_1_3_C	100	1	34	34	0	150.64	1	34	576.63	34	130.21	34	0.63546	1
N2C1W4_CL1_1_3_D	100	1	35	35	0.00%	187.87	1	35	646.27	35	126.46	35	0.4393	1
N2C1W4_CL1_1_3_E	100	1	33	33	0.00%	150.44	1	33	765.72	33	169.54	33	0.81841	1
N2C2W1_CL1_1_3_A	100	1	22	21	4.55%	3600.04	0	21	963.36	21	269.61	22	0.4736	0
N2C2W1_CL1_1_3_B	100	1	24	24	0.00%	3074.15	1	24	896.92	24	226.66	25	0.30763	0
N2C2W1_CL1_1_3_C	100	1	21	21	0.00%	0.00	1	21	984.33	21	290.33	21	0.29503	1
N2C2W1_CL1_1_3_D	100	1	22	22	0.00%	1.87	1	22	869.11	22	253.06	22	0.43913	1
N2C2W1_CL1_1_3_E	100	1	21	21	0.00%	0.05	1	21	993.33	21	273.54	21	0.27942	1
N2C2W2_CL1_1_3_A	100	1	26	26	0.00%	406.84	1	26	807.5	26	226.45	26	0.4785	1
N2C2W2_CL1_1_3_B	100	1	28	28	0.00%	473.18	1	27	768.14	27	206.15	28	0.43872	0
N2C2W2_CL1_1_3_C	100	1	27	27	0.00%	0.00	1	27	720.77	27	193.44	27	0.47817	1
N2C2W2_CL1_1_3_D	100	1	26	26	0.00%	0.00	1	26	756.38	26	213.78	26	0.52385	1
N2C2W2_CL1_1_3_E	100	1	27	26	3.70%	3600.03	0	26	755.1	26	208.05	27	0.45686	0
N2C2W4_CL1_1_3_A	100	1	28	27	3.57%	3600.03	0	27	831.78	27	204.42	28	0.53504	0
N2C2W4_CL1_1_3_B	100	1	29	29	0.00%	504.91	1	28	784.06	28	181.89	29	0.56674	0
N2C2W4_CL1_1_3_C	100	1	30	30	0.00%	150.82	1	30	637.86	30	159.57	30	0.50628	1
N2C2W4_CL1_1_3_D	100	1	29	29	0.00%	206.13	1	29	778.94	29	189.11	29	0.35499	1
N2C2W4_CL1_1_3_E	100	1	29	29	0.00%	164.89	1	29	707.75	29	173.66	29	0.58718	1
N2C3W1_CL1_1_3_A	100	1	20	20	0.00%	0.00	1	20	1101.7	20	278.88	20	0.33399	1
N2C3W1_CL1_1_3_B	100	1	19	19	0.00%	0.03	1	19	1108.3	19	288.05	19	0.31925	1
N2C3W1_CL1_1_3_C	100	1	19	19	0.00%	1320.76	1	19	1068.2	19	314.33	20	0.37462	0
N2C3W1_CL1_1_3_D	100	1	21	20	4.76%	3600.02	0	20	884.17	20	287.04	20	0.30699	1
N2C3W1_CL1_1_3_E	100	1	19	19	0.00%	0.00	1	19	1149.9	19	352.93	19	0.31661	1
N2C3W2_CL1_1_3_A	100	1	23	23	0.00%	151.46	1	23	939.55	23	279.56	23	0.44251	1
N2C3W2_CL1_1_3_B	100	1	23	23	0.00%	2057.25	1	23	842.7	23	251.04	23	0.34911	1
N2C3W2_CL1_1_3_C	100	1	23	23	0.00%	151.66	1	23	845.36	23	271.55	23	0.33827	1
N2C3W2_CL1_1_3_D	100	1	23	23	0.00%	0.00	1	23	1048.2	23	255.77	23	0.60592	1
N2C3W2_CL1_1_3_E	100	1	22	22	0.00%	0.00	1	22	1013.3	22	280.1	22	0.36465	1
N2C3W4_CL1_1_3_A	100	1	24	24	0.00%	28.11	1	24	916.06	24	242.87	24	0.42085	1
N2C3W4_CL1_1_3_B	100	1	25	25	0.00%	151.17	1	25	864.54	25	239.57	25	0.37415	1
N2C3W4_CL1_1_3_C	100	1	24	23	4.17%	3600.02	0	23	868.41	23	256.52	23	0.32644	1
N2C3W4_CL1_1_3_D	100	1	25	24	4.00%	3600.03	0	24	888.87	24	241.6	24	0.5025	1
N2C3W4_CL1_1_3_E	100	1	26	26	0.00%	200.52	1	26	760.74	26	227.37	27	0.39405	0

Table A.7: Detailed Comparison for N100 Class1

			VNS+CM					Lagrangian+GH						
Instance	**n**	**cl**	**UB**	**LB**	**gap**	**sec**	**opt**	**LB_lag1**	**sec**	**LB_lag2**	**sec**	**GH_UB**	**sec**	**opt**
N2C1W1_CL1_1_5_A	100	2	16	16	0.00%	0.00	1	16	1454.8	16	496.5	16	0.25739	1
N2C1W1_CL1_1_5_B	100	2	16	16	0.00%	0.00	1	16	1137.9	16	413.24	16	0.23927	1
N2C1W1_CL1_1_5_C	100	2	16	16	0.00%	0.00	1	15	1292.7	15	581.68	16	0.16592	0
N2C1W1_CL1_1_5_D	100	2	17	16	5.88%	3600.02	0	16	1131.9	16	478.9	17	0.26106	0
N2C1W1_CL1_1_5_E	100	2	18	18	0.00%	0.00	1	18	1033.2	18	319.07	18	0.3548	1
N2C1W2_CL1_1_5_A	100	2	21	21	0.00%	50.52	1	21	969.64	21	313.57	21	0.34177	1
N2C1W2_CL1_1_5_B	100	2	19	19	0.00%	1154.70	1	19	1025.1	19	401.23	20	0.39712	0
N2C1W2_CL1_1_5_C	100	2	20	20	0.00%	54.17	1	20	924.53	20	315.03	20	0.3419	1
N2C1W2_CL1_1_5_D	100	2	22	22	0.00%	0.00	1	22	884.49	22	286.29	22	0.51763	1
N2C1W2_CL1_1_5_E	100	2	22	22	0.00%	151.57	1	22	843.49	22	330.81	22	0.39925	1
N2C1W4_CL1_1_5_A	100	2	22	22	0.00%	0.00	1	22	848.77	22	272.83	22	0.39025	1
N2C1W4_CL1_1_5_B	100	2	22	22	0.00%	0.00	1	22	966.34	22	300.41	22	0.32764	1
N2C1W4_CL1_1_5_C	100	2	24	23	4.17%	3600.04	0	23	843.4	23	222.25	23	0.38068	1
N2C1W4_CL1_1_5_D	100	2	24	24	0.00%	151.39	1	24	998.18	24	244.2	24	0.37345	1
N2C1W4_CL1_1_5_E	100	2	23	23	0.00%	151.31	1	23	1003.3	23	247.68	23	0.40898	1
N2C2W1_CL1_1_5_A	100	2	15	15	0.00%	0.00	1	15	1413.5	15	488.21	15	0.24036	1
N2C2W1_CL1_1_5_B	100	2	17	17	0.00%	0.00	1	17	1152.6	17	408.02	17	0.23737	1
N2C2W1_CL1_1_5_C	100	2	15	15	0.00%	0.00	1	15	1544.9	15	576.2	15	0.18132	1
N2C2W1_CL1_1_5_D	100	2	16	15	6.25%	3600.02	0	15	1331.9	15	477.76	16	0.18209	0
N2C2W1_CL1_1_5_E	100	2	15	15	0.00%	0.00	1	15	1560.8	15	598.5	15	0.24655	1
N2C2W2_CL1_1_5_A	100	2	18	18	0.00%	0.26	1	18	1192.1	18	378.34	18	0.36971	1
N2C2W2_CL1_1_5_B	100	2	19	19	0.00%	0.06	1	19	1045.4	19	361.25	19	0.25827	1
N2C2W2_CL1_1_5_C	100	2	19	19	0.00%	0.00	1	19	1280	19	398.74	19	0.37462	1
N2C2W2_CL1_1_5_D	100	2	18	18	0.00%	0.00	1	18	1204.8	18	395.86	18	0.46152	1
N2C2W2_CL1_1_5_E	100	2	18	18	0.00%	171.63	1	18	1032.4	18	400.17	19	0.61682	0
N2C2W4_CL1_1_5_A	100	2	19	19	0.00%	0.00	1	19	998.33	19	356.66	19	0.40632	1
N2C2W4_CL1_1_5_B	100	2	20	20	0.00%	0.00	1	20	1040.7	20	338.72	20	0.54821	1
N2C2W4_CL1_1_5_C	100	2	21	21	0.00%	151.56	1	21	1196.5	21	297.07	21	0.50531	1
N2C2W4_CL1_1_5_D	100	2	20	20	0.00%	194.93	1	20	1042.5	20	292.49	20	0.4224	1
N2C2W4_CL1_1_5_E	100	2	20	20	0.00%	0.03	1	20	1043.7	20	355.73	20	0.43109	1
N2C3W1_CL1_1_5_A	100	2	14	14	0.00%	0.00	1	13	1375.9	13	509.09	14	0.23342	0
N2C3W1_CL1_1_5_B	100	2	13	13	0.00%	0.00	1	13	1426	13	584.29	13	0.18353	1
N2C3W1_CL1_1_5_C	100	2	13	13	0.00%	277.30	1	13	1384	13	524.8	14	0.21629	0
N2C3W1_CL1_1_5_D	100	2	14	14	0.00%	0.00	1	14	1549.3	14	583.69	14	0.28949	1
N2C3W1_CL1_1_5_E	100	2	13	13	0.00%	0.00	1	13	1540.7	13	830.21	13	0.18575	1
N2C3W2_CL1_1_5_A	100	2	16	16	0.00%	158.33	1	15	1299	15	568.43	16	0.23571	0
N2C3W2_CL1_1_5_B	100	2	16	16	0.00%	0.00	1	16	1430.8	16	492.79	16	0.34479	1
N2C3W2_CL1_1_5_C	100	2	16	16	0.00%	0.00	1	15	1233.4	15	510.45	16	0.42605	0
N2C3W2_CL1_1_5_D	100	2	16	16	0.00%	0.00	1	16	1487.6	16	516.36	16	0.46647	1
N2C3W2_CL1_1_5_E	100	2	15	15	0.00%	0.00	1	15	1415.6	15	564.55	15	0.66872	1
N2C3W4_CL1_1_5_A	100	2	17	17	0.00%	185.17	1	16	1380	16	440.65	17	0.5122	0
N2C3W4_CL1_1_5_B	100	2	17	17	0.00%	0.00	1	17	1317.3	17	461.47	17	0.41261	1
N2C3W4_CL1_1_5_C	100	2	16	16	0.00%	0.00	1	16	1339.4	16	555.98	16	0.42876	1
N2C3W4_CL1_1_5_D	100	2	17	17	0.00%	154.38	1	16	1281.9	16	531	17	0.42131	0
N2C3W4_CL1_1_5_E	100	2	16	16	0.00%	0.00	1	16	1332	16	540.46	16	0.24324	1

Table A.8: Detailed Comparison for N100 Class2

			VNS+CM					Lagrangian+GH						
Instance	**n**	**cl**	**UB**	**LB**	**gap**	**sec**	**opt**	**LB_lag1**	**sec**	**LB_lag2**	**sec**	**GH_UB**	**sec**	**opt**
N2C1W1_CL2_1_5_A	100	3	23	23	0.00%	0.00	1	23	765.91	23	230.71	23	0.46535	1
N2C1W1_CL2_1_5_B	100	3	22	22	0.00%	0.00	1	22	960.76	22	236.12	22	0.47871	1
N2C1W1_CL2_1_5_C	100	3	22	22	0.00%	0.68	1	22	794.91	22	313.38	22	0.50393	1
N2C1W1_CL2_1_5_D	100	3	23	23	0.00%	0.18	1	23	723.29	23	250.99	23	0.44934	1
N2C1W1_CL2_1_5_E	100	3	23	23	0.00%	152.66	1	23	793.46	23	203.55	24	0.5795	0
N2C1W2_CL2_1_5_A	100	3	26	26	0.00%	0.85	1	26	640.85	26	199	26	0.60448	1
N2C1W2_CL2_1_5_B	100	3	26	26	0.00%	150.82	1	26	773.19	26	238.34	26	0.5127	1
N2C1W2_CL2_1_5_C	100	3	26	26	0.00%	3.02	1	26	690.42	26	224.33	27	0.62502	0
N2C1W2_CL2_1_5_D	100	3	29	29	0.00%	151.50	1	29	636.99	29	129.69	30	0.74651	0
N2C1W2_CL2_1_5_E	100	3	26	26	0.00%	1119.22	1	26	718.2	26	183.65	27	0.62163	0
N2C1W4_CL2_1_5_A	100	3	29	29	0.00%	150.58	1	29	609.4	29	180.42	29	0.614	1
N2C1W4_CL2_1_5_B	100	3	27	27	0.00%	169.96	1	27	673.19	27	242.73	28	0.80072	0
N2C1W4_CL2_1_5_C	100	3	29	29	0.00%	172.54	1	28	589.73	28	193.18	29	0.58936	0
N2C1W4_CL2_1_5_D	100	3	29	29	0.00%	150.58	1	29	543	29	202.5	29	0.54323	1
N2C1W4_CL2_1_5_E	100	3	28	28	0.00%	150.64	1	28	719.86	28	208.32	28	0.73752	1
N2C2W1_CL2_1_5_A	100	3	22	21	4.55%	3600.04	0	21	811.53	21	222.76	22	0.35328	0
N2C2W1_CL2_1_5_B	100	3	23	23	0.00%	0.30	1	23	791.41	23	231.58	24	0.34453	0
N2C2W1_CL2_1_5_C	100	3	21	21	0.00%	156.53	1	21	834.24	21	215.5	22	0.36241	0
N2C2W1_CL2_1_5_D	100	3	22	22	0.00%	0.61	1	22	906.37	22	224.6	23	0.43518	0
N2C2W1_CL2_1_5_E	100	3	21	21	0.00%	151.82	1	21	863.57	21	237.72	21	0.37297	1
N2C2W2_CL2_1_5_A	100	3	25	25	0.00%	674.06	1	24	772.86	24	277.91	25	0.609	0
N2C2W2_CL2_1_5_B	100	3	25	25	0.00%	83.40	1	25	722.63	25	215.04	25	0.67451	1
N2C2W2_CL2_1_5_C	100	3	25	25	0.00%	0.13	1	25	804.07	25	214.65	25	0.42702	1
N2C2W2_CL2_1_5_D	100	3	24	24	0.00%	158.78	1	24	686.25	24	224.14	24	0.47111	1
N2C2W2_CL2_1_5_E	100	3	25	25	0.00%	151.08	1	24	764.94	24	226.74	25	0.41837	0
N2C2W4_CL2_1_5_A	100	3	25	25	0.00%	158.10	1	25	761.09	25	190.55	25	0.80853	1
N2C2W4_CL2_1_5_B	100	3	26	26	0.00%	150.92	1	26	673.07	26	186.9	26	0.71602	1
N2C2W4_CL2_1_5_C	100	3	26	26	0.00%	152.18	1	26	678.26	26	212.08	26	0.74182	1
N2C2W4_CL2_1_5_D	100	3	26	26	0.00%	0.00	1	26	694.16	26	259.13	26	0.42037	1
N2C2W4_CL2_1_5_E	100	3	26	26	0.00%	128.76	1	26	685.76	26	231.65	26	0.73308	1
N2C3W1_CL2_1_5_A	100	3	20	20	0.00%	0.00	1	20	977.9	20	231.35	20	0.54885	1
N2C3W1_CL2_1_5_B	100	3	20	20	0.00%	0.00	1	20	928.47	20	260.52	20	0.35513	1
N2C3W1_CL2_1_5_C	100	3	20	20	0.00%	156.60	1	20	923.95	20	251.26	21	0.50575	0
N2C3W1_CL2_1_5_D	100	3	21	21	0.00%	0.00	1	21	805.7	21	233.8	21	0.39806	1
N2C3W1_CL2_1_5_E	100	3	20	20	0.00%	0.00	1	20	931.66	20	261.66	20	0.288	1
N2C3W2_CL2_1_5_A	100	3	22	22	0.00%	668.57	1	22	890.87	22	269.36	22	0.3579	1
N2C3W2_CL2_1_5_B	100	3	23	23	0.00%	151.24	1	23	756.82	23	294.67	23	0.57603	1
N2C3W2_CL2_1_5_C	100	3	22	22	0.00%	155.17	1	22	838.67	22	253.85	22	0.42015	1
N2C3W2_CL2_1_5_D	100	3	23	22	4.35%	3600.02	0	22	855.2	22	221.47	23	0.57234	0
N2C3W2_CL2_1_5_E	100	3	22	22	0.00%	0.02	1	22	800.85	22	273.49	22	0.29313	1
N2C3W4_CL2_1_5_A	100	3	23	23	0.00%	152.49	1	23	804.05	23	224.58	23	0.61541	1
N2C3W4_CL2_1_5_B	100	3	23	23	0.00%	2110.43	1	23	804.63	23	219.79	24	0.69248	0
N2C3W4_CL2_1_5_C	100	3	23	23	0.00%	151.32	1	22	841.73	22	273.02	23	0.39731	0
N2C3W4_CL2_1_5_D	100	3	23	23	0.00%	152.43	1	23	798.3	23	264.78	23	0.40495	1
N2C3W4_CL2_1_5_E	100	3	23	22	4.35%	3600.02	0	22	762.44	22	257.63	22	0.54476	1

Table A.9: Detailed Comparison for N100 Class3

			VNS+CM					Lagrangian+GH						
Instance	n	cl	UB	LB	gap	sec	opt	LB_lag1	sec	LB_lag2	sec	GH_UB	sec	opt
N2C1W1_CL2_3_3_A	100	4	22	22	0.00%	51.70	1	22	766.21	22	279.21	23	0.43469	0
N2C1W1_CL2_3_3_B	100	4	22	22	0.00%	151.86	1	22	809.81	22	253.92	23	0.37923	0
N2C1W1_CL2_3_3_C	100	4	22	22	0.00%	0.06	1	22	723.45	22	248.39	22	0.50146	1
N2C1W1_CL2_3_3_D	100	4	23	23	0.00%	230.88	1	22	797.84	22	238.34	23	0.5641	0
N2C1W1_CL2_3_3_E	100	4	24	24	0.00%	0.52	1	24	835.15	24	263.39	24	0.39501	1
N2C1W2_CL2_3_3_A	100	4	26	26	0.00%	0.28	1	26	773.83	26	201.23	26	0.49306	1
N2C1W2_CL2_3_3_B	100	4	26	26	0.00%	151.18	1	26	628.16	26	232.93	26	0.48548	1
N2C1W2_CL2_3_3_C	100	4	26	26	0.00%	109.18	1	26	697.05	26	261.67	27	0.46285	0
N2C1W2_CL2_3_3_D	100	4	27	27	0.00%	667.19	1	27	676.88	27	220.66	28	1.3007	0
N2C1W2_CL2_3_3_E	100	4	27	27	0.00%	806.38	1	26	815.86	26	218.28	27	0.73912	0
N2C1W4_CL2_3_3_A	100	4	27	27	0.00%	410.71	1	27	728.22	27	264.03	28	0.64822	0
N2C1W4_CL2_3_3_B	100	4	27	27	0.00%	165.84	1	27	582.56	27	203.43	28	0.42822	0
N2C1W4_CL2_3_3_C	100	4	29	29	0.00%	150.64	1	29	513.94	29	152.62	29	0.4916	1
N2C1W4_CL2_3_3_D	100	4	29	29	0.00%	150.72	1	29	561.08	29	216.43	29	0.41082	1
N2C1W4_CL2_3_3_E	100	4	28	28	0.00%	150.74	1	28	491.4	28	154.18	28	0.46726	1
N2C2W1_CL2_3_3_A	100	4	20	20	0.00%	4.60	1	20	879.88	20	269.64	20	0.44903	1
N2C2W1_CL2_3_3_B	100	4	22	22	0.00%	152.95	1	22	893.82	22	265.33	23	0.53263	0
N2C2W1_CL2_3_3_C	100	4	20	20	0.00%	0.02	1	20	1081.5	20	328.3	20	0.68776	1
N2C2W1_CL2_3_3_D	100	4	21	21	0.00%	0.52	1	21	846.29	21	261.51	22	0.43824	0
N2C2W1_CL2_3_3_E	100	4	20	20	0.00%	0.00	1	20	958.49	20	282.74	20	0.42446	1
N2C2W2_CL2_3_3_A	100	4	24	24	0.00%	150.90	1	24	762.06	24	245.07	24	0.39925	1
N2C2W2_CL2_3_3_B	100	4	25	25	0.00%	168.07	1	24	713.38	24	239.88	25	0.61236	0
N2C2W2_CL2_3_3_C	100	4	24	24	0.00%	151.72	1	24	691.87	24	258.41	24	0.6111	1
N2C2W2_CL2_3_3_D	100	4	24	24	0.00%	151.10	1	24	647.75	24	264.04	24	0.71522	1
N2C2W2_CL2_3_3_E	100	4	24	24	0.00%	0.27	1	24	641.12	24	242.1	24	0.46677	1
N2C2W4_CL2_3_3_A	100	4	25	25	0.00%	157.39	1	24	709.73	24	306.67	25	0.75598	0
N2C2W4_CL2_3_3_B	100	4	25	25	0.00%	12.34	1	25	668.24	25	340.01	25	1.5336	1
N2C2W4_CL2_3_3_C	100	4	26	26	0.00%	152.27	1	25	720.2	25	249	26	1.0307	0
N2C2W4_CL2_3_3_D	100	4	25	25	0.00%	184.11	1	25	690.15	25	286.34	26	1.1318	0
N2C2W4_CL2_3_3_E	100	4	25	25	0.00%	3049.97	1	25	645.78	25	262.69	26	1.077	0
N2C3W1_CL2_3_3_A	100	4	19	19	0.00%	0.00	1	19	947.59	19	482.96	19	0.62335	1
N2C3W1_CL2_3_3_B	100	4	19	19	0.00%	1468.11	1	18	911.63	18	440.29	19	0.63705	0
N2C3W1_CL2_3_3_C	100	4	19	19	0.00%	0.00	1	19	922.06	19	462.33	19	0.46842	1
N2C3W1_CL2_3_3_D	100	4	20	20	0.00%	170.23	1	19	719.23	19	518.14	19	0.98559	1
N2C3W1_CL2_3_3_E	100	4	19	18	5.26%	3600.03	0	18	953.7	18	475.31	19	0.74751	0
N2C3W2_CL2_3_3_A	100	4	21	21	0.00%	0.03	1	21	926.93	21	395.14	21	0.65956	1
N2C3W2_CL2_3_3_B	100	4	22	22	0.00%	151.42	1	21	849.54	21	459.39	22	0.77548	0
N2C3W2_CL2_3_3_C	100	4	21	21	0.00%	27.60	1	21	724.64	21	433.93	21	0.64588	1
N2C3W2_CL2_3_3_D	100	4	22	21	4.55%	3600.03	0	21	767.43	21	364.61	22	1.0645	0
N2C3W2_CL2_3_3_E	100	4	21	21	0.00%	0.00	1	21	811.23	21	419.35	21	0.93214	1
N2C3W4_CL2_3_3_A	100	4	22	22	0.00%	152.33	1	22	789	22	250.29	22	0.57728	1
N2C3W4_CL2_3_3_B	100	4	23	23	0.00%	167.44	1	22	741.76	22	235.14	23	0.54523	0
N2C3W4_CL2_3_3_C	100	4	22	22	0.00%	151.23	1	22	731.55	22	277.66	22	0.36033	1
N2C3W4_CL2_3_3_D	100	4	22	22	0.00%	161.12	1	22	766.42	22	281.31	22	0.70075	1
N2C3W4_CL2_3_3_E	100	4	23	23	0.00%	151.25	1	23	723.23	23	252.29	23	0.3721	1

Table A.10: Detailed Comparison for N100 Class4

			VNS+CM					Lagrangian+GH						
Instance	n	cl	UB	LB	gap	sec	opt	LB_lag1	sec	LB_lag2	sec	GH_UB	sec	opt
N2C1W1_CL2_3_4_A	100	5	19	19	0.00%	0.00	1	19	887.81	19	375.22			0
N2C1W1_CL2_3_4_B	100	5	20	20	0.00%	0.00	1	20	922.8	20	351.46	20	0.30173	1
N2C1W1_CL2_3_4_C	100	5	19	19	0.00%	0.00	1	19	990.66	19	342.09	19	0.26937	1
N2C1W1_CL2_3_4_D	100	5	20	20	0.00%	151.78	1	19	930.87	19	307.3	20	0.34581	0
N2C1W1_CL2_3_4_E	100	5	21	21	0.00%	69.58	1	21	922.42	21	258.7	22	0.52527	0
N2C1W2_CL2_3_4_A	100	5	24	23	4.17%	3600.03	0	23	759.44	23	273.66	24	0.414	0
N2C1W2_CL2_3_4_B	100	5	22	22	0.00%	0.00	1	22	874.77	22	280.01	22	0.64256	1
N2C1W2_CL2_3_4_C	100	5	23	23	0.00%	257.23	1	22	760.89	22	242.54	22	0.437	1
N2C1W2_CL2_3_4_D	100	5	24	24	0.00%	43.56	1	24	720.69	24	231.07	24	0.57986	1
N2C1W2_CL2_3_4_E	100	5	24	24	0.00%	151.10	1	24	684.65	24	239.7	24	0.54237	1
N2C1W4_CL2_3_4_A	100	5	24	24	0.00%	261.08	1	24	717.3	24	237.47	25	0.61448	0
N2C1W4_CL2_3_4_B	100	5	23	23	0.00%	829.04	1	23	814.28	23	262.25	23	0.64638	1
N2C1W4_CL2_3_4_C	100	5	24	24	0.00%	0.00	1	24	872.1	24	234.12	24	0.49966	1
N2C1W4_CL2_3_4_D	100	5	24	24	0.00%	681.63	1	24	689.55	24	211.92	24	0.693	1
N2C1W4_CL2_3_4_E	100	5	24	24	0.00%	154.64	1	23	717.29	23	244.31	24	0.49765	0
N2C2W1_CL2_3_4_A	100	5	18	18	0.00%	0.00	1	18	1018.9	18	427.2	18	0.29493	1
N2C2W1_CL2_3_4_B	100	5	19	19	0.00%	5.22	1	19	952.54	19	337.42	20	0.38062	0
N2C2W1_CL2_3_4_C	100	5	18	18	0.00%	151.88	1	17	1100.3	17	407.92	18	0.31011	0
N2C2W1_CL2_3_4_D	100	5	18	18	0.00%	20.48	1	18	1103.8	18	410.63	19	0.52716	0
N2C2W1_CL2_3_4_E	100	5	17	17	0.00%	2489.49	1	17	1318.7	17	388.46	18	0.36899	0
N2C2W2_CL2_3_4_A	100	5	21	20	4.76%	3600.03	0	20	1038.7	20	290.81	21	0.43277	0
N2C2W2_CL2_3_4_B	100	5	21	21	0.00%	38.91	1	21	894.88	21	270.1	22	0.61609	0
N2C2W2_CL2_3_4_C	100	5	21	21	0.00%	151.96	1	21	925.51	21	268.69	21	0.63773	1
N2C2W2_CL2_3_4_D	100	5	21	20	4.76%	3600.03	0	20	1030.1	20	324.11	20	0.60467	1
N2C2W2_CL2_3_4_E	100	5	21	21	0.00%	153.93	1	20	882.04	20	320.8	21	0.40148	0
N2C2W4_CL2_3_4_A	100	5	21	21	0.00%	0.00	1	21	967.24	21	270.97	21	0.63742	1
N2C2W4_CL2_3_4_B	100	5	21	21	0.00%	710.24	1	21	918.01	21	262.36	22	0.4853	0
N2C2W4_CL2_3_4_C	100	5	22	22	0.00%	6.60	1	22	892.84	22	264.34	22	0.60286	1
N2C2W4_CL2_3_4_D	100	5	22	22	0.00%	151.19	1	22	769.44	22	304.23	22	0.42056	1
N2C2W4_CL2_3_4_E	100	5	22	22	0.00%	151.11	1	22	855.39	22	289.4	22	0.79305	1
N2C3W1_CL2_3_4_A	100	5	17	17	0.00%	157.28	1	16	1064.6	16	394.96	17	0.35631	0
N2C3W1_CL2_3_4_B	100	5	17	16	5.88%	3600.03	0	16	1065.2	16	437.07	16	0.35889	1
N2C3W1_CL2_3_4_C	100	5	17	17	0.00%	349.53	1	16	984.01	16	419.43	17	0.22408	0
N2C3W1_CL2_3_4_D	100	5	17	17	0.00%	0.00	1	17	991.86	17	413.58	17	0.26792	1
N2C3W1_CL2_3_4_E	100	5	16	16	0.00%	1051.71	1	16	1146.5	16	523.28	16	0.47714	1
N2C3W2_CL2_3_4_A	100	5	18	18	0.00%	642.86	1	18	960.82	18	389.64	19	0.42306	0
N2C3W2_CL2_3_4_B	100	5	19	19	0.00%	152.27	1	19	1005.5	19	367.53	19	0.42955	1
N2C3W2_CL2_3_4_C	100	5	18	18	0.00%	440.17	1	18	944.59	18	366.04	18	0.28834	1
N2C3W2_CL2_3_4_D	100	5	19	18	5.26%	3600.03	0	18	914.59	18	352.8	19	0.3587	0
N2C3W2_CL2_3_4_E	100	5	18	18	0.00%	0.00	1	18	1098.7	18	429.38	18	0.67708	1
N2C3W4_CL2_3_4_A	100	5	19	19	0.00%	0.00	1	19	974.55	19	348.06	19	0.557	1
N2C3W4_CL2_3_4_B	100	5	19	19	0.00%	221.00	1	19	1112.8	19	306.98	20	0.57607	0
N2C3W4_CL2_3_4_C	100	5	19	19	0.00%	152.05	1	19	971.69	19	422.03	19	0.4494	1
N2C3W4_CL2_3_4_D	100	5	19	19	0.00%	0.12	1	19	961.56	19	366.09	20	0.38341	0
N2C3W4_CL2_3_4_E	100	5	20	20	0.00%	151.87	1	20	880.74	20	352.68	20	0.45185	1

Table A.11: Detailed Comparison for N100 Class5

			VNS+CM							GH		
Instance	**n**	**cl**	**UB_0**	**L$_f$**	**LB**	**UB**	**gap**	**sec**	**opt**	**GH_UB**	**sec**	**opt**
N3C1W1_CL1_1_3_A	200	1	51	50	51	50	1.96%	3600.00	0	50	1.3954	1
N3C1W1_CL1_1_3_B	200	1	53	52	53	52	1.89%	3600.00	0	52	0.89029	1
N3C1W1_CL1_1_3_C	200	1	48	47	48	47	2.08%	3600.00	0	48	1.0029	0
N3C1W1_CL1_1_3_D	200	1	53	51	53	51	3.77%	3600.00	0	52	0.87539	0
N3C1W1_CL1_1_3_E	200	1	47	47	47	47	0.00%	0.02	1	47	0.97052	1
N3C1W2_CL1_1_3_A	200	1	58	56	58	56	3.45%	3600.00	0	57	1.5874	0
N3C1W2_CL1_1_3_B	200	1	60	58	60	58	3.33%	3600.00	0	59	1.4629	0
N3C1W2_CL1_1_3_C	200	1	60	58	60	58	3.33%	3600.00	0	59	1.1186	0
N3C1W2_CL1_1_3_D	200	1	63	61	63	61	3.17%	3600.00	0	62	1.3386	0
N3C1W2_CL1_1_3_E	200	1	61	58	61	58	4.92%	3600.00	0	59	1.6245	0
N3C1W4_CL1_1_3_A	200	1	68	64	68	64	5.88%	3600.00	0	66	1.3432	0
N3C1W4_CL1_1_3_B	200	1	65	62	65	62	4.62%	3600.00	0	64	1.8039	0
N3C1W4_CL1_1_3_C	200	1	64	62	64	62	3.13%	3600.00	0	63	1.4182	0
N3C1W4_CL1_1_3_D	200	1	69	66	69	66	4.35%	3600.00	0	68	1.1904	0
N3C1W4_CL1_1_3_E	200	1	67	64	67	64	4.48%	3600.00	0	65	1.579	0
N3C2W1_CL1_1_3_A	200	1	48	47	48	47	2.08%	3600.00	0	47	2.1683	1
N3C2W1_CL1_1_3_B	200	1	44	44	44	44	0.00%	0.01	1	44	1.7024	1
N3C2W1_CL1_1_3_C	200	1	43	42	43	42	2.33%	3600.00	0	43	3.3787	0
N3C2W1_CL1_1_3_D	200	1	43	42	43	42	2.33%	3600.00	0	42	1.2465	1
N3C2W1_CL1_1_3_E	200	1	44	43	44	43	2.27%	3600.00	0	44	1.5562	0
N3C2W2_CL1_1_3_A	200	1	53	52	53	52	1.89%	3600.00	0	52	1.8886	1
N3C2W2_CL1_1_3_B	200	1	51	50	51	50	1.96%	3600.00	0	50	2.0488	1
N3C2W2_CL1_1_3_C	200	1	52	51	52	51	1.92%	3600.00	0	52	2.1769	0
N3C2W2_CL1_1_3_D	200	1	53	52	53	52	1.89%	3600.00	0	53	2.2999	0
N3C2W2_CL1_1_3_E	200	1	56	54	56	54	3.57%	3600.00	0	55	1.9459	0
N3C2W4_CL1_1_3_A	200	1	56	54	56	54	3.57%	3600.00	0	55	1.8162	0
N3C2W4_CL1_1_3_B	200	1	55	53	55	53	3.64%	3600.00	0	54	2.1338	0
N3C2W4_CL1_1_3_C	200	1	60	58	60	58	3.33%	3600.00	0	59	2.0408	0
N3C2W4_CL1_1_3_D	200	1	56	54	56	54	3.57%	3600.00	0	55	1.9668	0
N3C2W4_CL1_1_3_E	200	1	55	53	55	53	3.64%	3600.00	0	53	2.0273	1
N3C3W1_CL1_1_3_A	200	1	35	35	35	35	0.00%	0.01	1	35	2.2495	1
N3C3W1_CL1_1_3_B	200	1	38	38	38	38	0.00%	0.01	1	38	4.2586	1
N3C3W1_CL1_1_3_C	200	1	37	37	37	37	0.00%	0.00	1	37	1.8482	1
N3C3W1_CL1_1_3_D	200	1	34	34	34	34	0.00%	0.00	1	34	1.5609	1
N3C3W1_CL1_1_3_E	200	1	37	36	37	36	2.70%	3600.00	0	36	1.7877	1
N3C3W2_CL1_1_3_A	200	1	44	43	44	44	0.00%	151.01	1	44	2.3485	1
N3C3W2_CL1_1_3_B	200	1	43	42	43	42	2.33%	3600.00	0	42	2.7072	1
N3C3W2_CL1_1_3_C	200	1	43	42	43	42	2.33%	3600.00	0	43	2.298	0
N3C3W2_CL1_1_3_D	200	1	42	41	42	41	2.38%	3600.00	0	41	2.4699	1
N3C3W2_CL1_1_3_E	200	1	42	41	42	41	2.38%	3600.00	0	41	2.6098	1
N3C3W4_CL1_1_3_A	200	1	48	46	47	46	2.13%	3600.00	0	47	3.0313	0
N3C3W4_CL1_1_3_B	200	1	47	45	47	45	4.26%	3600.00	0	46	2.8998	0
N3C3W4_CL1_1_3_C	200	1	47	46	47	46	2.13%	3600.00	0	46	3.0513	1
N3C3W4_CL1_1_3_D	200	1	47	45	47	45	4.26%	3600.00	0	46	2.6339	0
N3C3W4_CL1_1_3_E	200	1	46	44	46	44	4.35%	3600.00	0	45	2.5146	0

Table A.12: Detailed Comparison for N200 Class1

			VNS+CM							GH		
Instance	**n**	**cl**	**UB_0**	**L_f**	**LB**	**UB**	**gap**	**sec**	**opt**	**GH_UB**	**sec**	**opt**
N3C1W1_CL1_1_5_A	200	2	34	33	34	33	2.94%	3600.00	0	34	1.6783	0
N3C1W1_CL1_1_5_B	200	2	37	36	37	36	2.70%	3600.00	0	37	1.7891	0
N3C1W1_CL1_1_5_C	200	2	33	32	33	32	3.03%	3600.00	0	33	1.5306	0
N3C1W1_CL1_1_5_D	200	2	35	35	35	35	0.00%	0.02	1	35	1.5031	1
N3C1W1_CL1_1_5_E	200	2	32	32	32	32	0.00%	0.00	1	32	1.4075	1
N3C1W2_CL1_1_5_A	200	2	39	39	39	39	0.00%	0.25	1	39	2.5124	1
N3C1W2_CL1_1_5_B	200	2	38	37	38	38	0.00%	181.46	1	38	2.3419	1
N3C1W2_CL1_1_5_C	200	2	40	40	40	40	0.00%	1.16	1	40	2.7945	1
N3C1W2_CL1_1_5_D	200	2	42	42	42	42	0.00%	6.39	1	42	2.4234	1
N3C1W2_CL1_1_5_E	200	2	41	40	41	40	2.44%	3600.00	0	41	2.6864	0
N3C1W4_CL1_1_5_A	200	2	45	44	45	44	2.22%	3600.00	0	44	2.5358	1
N3C1W4_CL1_1_5_B	200	2	44	43	44	43	2.27%	3600.00	0	43	2.0625	1
N3C1W4_CL1_1_5_C	200	2	44	43	44	43	2.27%	3600.00	0	43	2.601	1
N3C1W4_CL1_1_5_D	200	2	44	43	44	43	2.27%	3600.00	0	44	3.112	0
N3C1W4_CL1_1_5_E	200	2	42	41	42	41	2.38%	3600.00	0	41	2.9466	1
N3C2W1_CL1_1_5_A	200	2	30	30	30	30	0.00%	0.01	1	30	1.496	1
N3C2W1_CL1_1_5_B	200	2	28	28	28	28	0.00%	0.00	1	28	1.8795	1
N3C2W1_CL1_1_5_C	200	2	29	29	29	29	0.00%	0.00	1	29	1.2735	1
N3C2W1_CL1_1_5_D	200	2	29	29	29	29	0.00%	0.01	1	29	1.7728	1
N3C2W1_CL1_1_5_E	200	2	30	29	30	29	3.33%	3600.00	0	30	1.5709	0
N3C2W2_CL1_1_5_A	200	2	36	35	36	35	2.78%	3600.00	0	36	2.1646	0
N3C2W2_CL1_1_5_B	200	2	35	34	35	34	2.86%	3600.00	0	35	2.1303	0
N3C2W2_CL1_1_5_C	200	2	35	34	35	35	0.00%	157.81	1	35	3.0828	1
N3C2W2_CL1_1_5_D	200	2	36	35	36	35	2.78%	3600.00	0	36	2.4474	0
N3C2W2_CL1_1_5_E	200	2	38	37	38	37	2.63%	3600.00	0	37	2.8712	1
N3C2W4_CL1_1_5_A	200	2	38	36	38	36	5.26%	3600.00	0	37	2.6973	0
N3C2W4_CL1_1_5_B	200	2	37	36	37	36	2.70%	3600.00	0	37	2.67	0
N3C2W4_CL1_1_5_C	200	2	40	39	40	39	2.50%	3600.00	0	40	2.0975	0
N3C2W4_CL1_1_5_D	200	2	38	37	38	37	2.63%	3600.00	0	37	3.1096	1
N3C2W4_CL1_1_5_E	200	2	37	36	37	36	2.70%	3600.00	0	36	5.4772	1
N3C3W1_CL1_1_5_A	200	2	24	24	24	24	0.00%	0.01	1	24	1.6847	1
N3C3W1_CL1_1_5_B	200	2	26	26	26	26	0.00%	0.00	1	26	3.9074	1
N3C3W1_CL1_1_5_C	200	2	25	25	25	25	0.00%	0.01	1	25	2.2657	1
N3C3W1_CL1_1_5_D	200	2	23	23	23	23	0.00%	0.00	1	23	1.7823	1
N3C3W1_CL1_1_5_E	200	2	25	25	25	25	0.00%	0.00	1	25	2.2083	1
N3C3W2_CL1_1_5_A	200	2	29	29	29	29	0.00%	0.00	1	29	3.4058	1
N3C3W2_CL1_1_5_B	200	2	29	28	29	28	3.45%	3600.00	0	29	3.7448	0
N3C3W2_CL1_1_5_C	200	2	29	28	29	28	3.45%	3600.00	0	29	3.037	0
N3C3W2_CL1_1_5_D	200	2	28	27	28	28	0.00%	153.03	1	28	1.7521	1
N3C3W2_CL1_1_5_E	200	2	28	27	28	28	0.00%	173.33	1	28	2.0183	1
N3C3W4_CL1_1_5_A	200	2	31	31	31	31	0.00%	0.00	1	31	2.4742	1
N3C3W4_CL1_1_5_B	200	2	31	30	31	30	3.23%	3600.00	0	30	4.5288	1
N3C3W4_CL1_1_5_C	200	2	31	30	31	31	0.00%	157.32	1	31	3.4009	1
N3C3W4_CL1_1_5_D	200	2	31	30	31	30	3.23%	3600.00	0	31	3.9596	0
N3C3W4_CL1_1_5_E	200	2	30	30	30	30	0.00%	0.01	1	30	3.8927	1

Table A.13: Detailed Comparison for N200 Class2

			VNS+CM							GH		
Instance	n	cl	UB_0	L_f	LB	UB	gap	sec	opt	GH_UB	sec	opt
N3C1W1_CL2_1_5_A	200	3	48	47	48	47	2.08%	3600.00	0	47	6.2278	1
N3C1W1_CL2_1_5_B	200	3	47	46	47	46	2.13%	3600.00	0	47	5.7661	0
N3C1W1_CL2_1_5_C	200	3	44	43	44	43	2.27%	3600.00	0	44	3.9557	0
N3C1W1_CL2_1_5_D	200	3	49	47	49	47	4.08%	3600.00	0	48	3.7521	0
N3C1W1_CL2_1_5_E	200	3	44	43	44	43	2.27%	3600.00	0	44	3.0251	0
N3C1W2_CL2_1_5_A	200	3	51	49	51	49	3.92%	3600.00	0	50	3.4482	0
N3C1W2_CL2_1_5_B	200	3	52	50	52	50	3.85%	3600.00	0	51	3.3281	0
N3C1W2_CL2_1_5_C	200	3	52	50	52	50	3.85%	3600.00	0	51	3.6665	0
N3C1W2_CL2_1_5_D	200	3	53	52	53	52	1.89%	3600.00	0	53	3.7915	0
N3C1W2_CL2_1_5_E	200	3	52	50	52	50	3.85%	3600.00	0	51	3.658	0
N3C1W4_CL2_1_5_A	200	3	56	54	56	54	3.57%	3600.00	0	56	3.8099	0
N3C1W4_CL2_1_5_B	200	3	54	53	54	53	1.85%	3600.00	0	54	3.8397	0
N3C1W4_CL2_1_5_C	200	3	54	53	54	53	1.85%	3600	0	53	3.8978	1
N3C1W4_CL2_1_5_D	200	3	56	54	56	54	3.57%	3600.00	0	56	3.7796	0
N3C1W4_CL2_1_5_E	200	3	55	53	55	53	3.64%	3600.00	0	54	3.6719	0
N3C2W1_CL2_1_5_A	200	3	43	43	43	43	0.00%	0.01	1	43	3.322	1
N3C2W1_CL2_1_5_B	200	3	41	41	41	41	0.00%	0.74	1	41	2.3048	1
N3C2W1_CL2_1_5_C	200	3	42	41	42	41	2.38%	3600.00	0	42	2.5449	0
N3C2W1_CL2_1_5_D	200	3	41	40	41	41	0.00%	154.821	1	41	2.6703	1
N3C2W1_CL2_1_5_E	200	3	42	41	42	41	2.38%	3600.00	0	41	2.2468	1
N3C2W2_CL2_1_5_A	200	3	48	47	48	47	2.08%	3600.00	0	47	3.335	1
N3C2W2_CL2_1_5_B	200	3	47	46	47	46	2.13%	3600.00	0	47	3.2222	0
N3C2W2_CL2_1_5_C	200	3	48	47	48	47	2.08%	3600.00	0	47	3.2401	1
N3C2W2_CL2_1_5_D	200	3	49	47	49	47	4.08%	3600.00	0	48	3.9272	0
N3C2W2_CL2_1_5_E	200	3	50	48	50	48	4.00%	3600.00	0	49	2.3207	0
N3C2W4_CL2_1_5_A	200	3	50	48	50	48	4.00%	3600.00	0	49	3.0419	0
N3C2W4_CL2_1_5_B	200	3	50	48	50	48	4.00%	3600.00	0	49	2.8939	0
N3C2W4_CL2_1_5_C	200	3	52	50	52	50	3.85%	3600.00	0	51	4.2761	0
N3C2W4_CL2_1_5_D	200	3	51	49	51	49	3.92%	3600.00	0	49	4.3326	1
N3C2W4_CL2_1_5_E	200	3	50	48	50	48	4.00%	3600.00	0	49	4.3029	0
N3C3W1_CL2_1_5_A	200	3	36	36	36	36	0.00%	0.01	1	36	2.107	1
N3C3W1_CL2_1_5_B	200	3	39	38	39	38	2.56%	3600.00	0	38	2.3335	1
N3C3W1_CL2_1_5_C	200	3	37	37	37	37	0.00%	0.01	1	37	2.2433	1
N3C3W1_CL2_1_5_D	200	3	35	35	35	35	0.00%	0.01	1	36	1.6192	0
N3C3W1_CL2_1_5_E	200	3	37	36	37	36	2.70%	3600.00	0	37	1.4299	0
N3C3W2_CL2_1_5_A	200	3	44	43	44	43	2.27%	3600.00	0	43	2.9107	1
N3C3W2_CL2_1_5_B	200	3	44	42	44	42	4.55%	3600.00	0	43	2.2512	0
N3C3W2_CL2_1_5_C	200	3	44	42	44	42	4.55%	3600.00	0	43	2.104	0
N3C3W2_CL2_1_5_D	200	3	43	42	43	42	2.33%	3600.00	0	42	2.2359	1
N3C3W2_CL2_1_5_E	200	3	43	41	43	41	4.65%	3600.00	0	42	3.5759	0
N3C3W4_CL2_1_5_A	200	3	46	44	46	44	4.35%	3600.00	0	45	3.1189	0
N3C3W4_CL2_1_5_B	200	3	46	44	46	44	4.35%	3600.00	0	45	3.8072	0
N3C3W4_CL2_1_5_C	200	3	46	44	46	44	4.35%	3600.00	0	45	2.8621	0
N3C3W4_CL2_1_5_D	200	3	46	44	46	44	4.35%	3600.00	0	45	4.552	0
N3C3W4_CL2_1_5_E	200	3	45	43	45	43	4.44%	3600.00	0	44	3.2726	0

Table A.14: Detailed Comparison for N200 Class3

			VNS+CM							GH		
Instance	**n**	**cl**	**UB_0**	**L$_f$**	**LB**	**UB**	**gap**	**sec**	**opt**	**GH_UB**	**sec**	**opt**
N3C1W1_CL2_3_3_A	200	4	45	45	45	45	0.00%	5.01	1	45	3.5447	1
N3C1W1_CL2_3_3_B	200	4	47	46	47	47	0.00%	735.31	1	47	3.7086	1
N3C1W1_CL2_3_3_C	200	4	45	44	45	44	2.22%	3600.00	0	45	2.8672	0
N3C1W1_CL2_3_3_D	200	4	48	47	48	47	2.08%	3600.00	0	47	2.6959	1
N3C1W1_CL2_3_3_E	200	4	44	43	44	43	2.27%	3600.00	0	44	2.7267	0
N3C1W2_CL2_3_3_A	200	4	52	50	52	50	3.85%	3600.00	0	51	3.3078	0
N3C1W2_CL2_3_3_B	200	4	51	50	51	50	1.96%	3600.00	0	51	3.8614	0
N3C1W2_CL2_3_3_C	200	4	53	51	53	51	3.77%	3600.00	0	52	3.5929	0
N3C1W2_CL2_3_3_D	200	4	54	52	54	52	3.70%	3600.00	0	53	4.1916	0
N3C1W2_CL2_3_3_E	200	4	53	51	53	51	3.77%	3600.00	0	52	4.1755	0
N3C1W4_CL2_3_3_A	200	4	56	54	56	54	3.57%	3600.00	0	56	4.7535	0
N3C1W4_CL2_3_3_B	200	4	55	54	55	54	1.82%	3600.00	0	55	3.9654	0
N3C1W4_CL2_3_3_C	200	4	55	53	55	53	3.64%	3600.00	0	55	3.2452	0
N3C1W4_CL2_3_3_D	200	4	56	54	56	54	3.57%	3600.00	0	56	2.9046	0
N3C1W4_CL2_3_3_E	200	4	55	53	55	53	3.64%	3600.00	0	54	4.7521	0
N3C2W1_CL2_3_3_A	200	4	43	42	43	42	2.33%	3600.00	0	42	2.8839	1
N3C2W1_CL2_3_3_B	200	4	41	40	41	41	0.00%	151.53	1	41	2.9316	1
N3C2W1_CL2_3_3_C	200	4	42	41	42	41	2.38%	3600.00	0	42	2.4658	0
N3C2W1_CL2_3_3_D	200	4	41	40	41	40	2.44%	3600.00	0	40	2.2986	1
N3C2W1_CL2_3_3_E	200	4	41	41	41	41	0.00%	0.00	1	41	2.9918	1
N3C2W2_CL2_3_3_A	200	4	48	47	48	47	2.08%	3600.00	0	47	3.1405	1
N3C2W2_CL2_3_3_B	200	4	47	46	47	46	2.13%	3600.00	0	46	3.2319	1
N3C2W2_CL2_3_3_C	200	4	47	46	47	46	2.13%	3600.00	0	47	3.1297	0
N3C2W2_CL2_3_3_D	200	4	48	47	48	47	2.08%	3600.00	0	47	3.9767	1
N3C2W2_CL2_3_3_E	200	4	49	48	49	48	2.04%	3600.00	0	49	2.9741	0
N3C2W4_CL2_3_3_A	200	4	50	48	50	48	4.00%	3600.00	0	49	3.157	0
N3C2W4_CL2_3_3_B	200	4	49	48	49	48	2.04%	3600	0	48	3.4726	1
N3C2W4_CL2_3_3_C	200	4	51	50	51	50	1.96%	3600.00	0	51	3.8268	0
N3C2W4_CL2_3_3_D	200	4	50	48	50	48	4.00%	3600.00	0	49	3.1797	0
N3C2W4_CL2_3_3_E	200	4	49	47	49	47	4.08%	3600.00	0	48	3.4501	0
N3C3W1_CL2_3_3_A	200	4	35	35	35	35	0.00%	0.00	1	35	2.0118	1
N3C3W1_CL2_3_3_B	200	4	38	37	38	38	0.00%	151.03	1	38	2.4217	1
N3C3W1_CL2_3_3_C	200	4	37	36	37	36	2.70%	3600.00	0	36	2.2571	1
N3C3W1_CL2_3_3_D	200	4	35	34	35	34	2.86%	3600.00	0	35	2.28	0
N3C3W1_CL2_3_3_E	200	4	36	36	36	36	0.00%	0.00	1	36	2.137	1
N3C3W2_CL2_3_3_A	200	4	43	42	43	42	2.33%	3600.00	0	43	2.7106	0
N3C3W2_CL2_3_3_B	200	4	42	41	42	42	0.00%	387.56	1	42	3.2675	1
N3C3W2_CL2_3_3_C	200	4	42	41	42	42	0.00%	154.42	1	42	2.9445	1
N3C3W2_CL2_3_3_D	200	4	42	40	42	40	4.76%	3600.00	0	41	2.9838	0
N3C3W2_CL2_3_3_E	200	4	41	40	41	41	0.00%	151.46	1	41	3.3888	1
N3C3W4_CL2_3_3_A	200	4	45	43	45	43	4.44%	3600.00	0	44	3.4652	0
N3C3W4_CL2_3_3_B	200	4	44	43	44	43	2.27%	3600.00	0	44	3.8784	0
N3C3W4_CL2_3_3_C	200	4	45	43	45	43	4.44%	3600.00	0	44	3.9578	0
N3C3W4_CL2_3_3_D	200	4	44	43	44	43	2.27%	3600.00	0	44	3.3832	0
N3C3W4_CL2_3_3_E	200	4	44	42	44	42	4.55%	3600.00	0	43	3.6386	0

Table A.15: Detailed Comparison for N200 Class4

			VNS+CM							GH		
Instance	**n**	**cl**	**UB_0**	**L_f**	**LB**	**UB**	**gap**	**sec**	**opt**	**GH_UB**	**sec**	**opt**
N3C1W1_CL2_3_4_A	200	5	40	39	40	40	0.00%	151.09	1	40	3.2183	1
N3C1W1_CL2_3_4_B	200	5	42	41	42	41	2.38%	3600.00	0	41	2.7475	1
N3C1W1_CL2_3_4_C	200	5	38	38	38	38	0.00%	0.00	1	38	1.9091	1
N3C1W1_CL2_3_4_D	200	5	40	40	40	40	0.00%	5.50	1	40	2.5552	1
N3C1W1_CL2_3_4_E	200	5	38	38	38	38	0.00%	69.61	1	38	2.1881	1
N3C1W2_CL2_3_4_A	200	5	45	44	45	44	2.22%	3600.00	0	44	3.3574	1
N3C1W2_CL2_3_4_B	200	5	45	44	45	44	2.22%	3600.00	0	44	3.395	1
N3C1W2_CL2_3_4_C	200	5	44	43	44	43	2.27%	3600.00	0	44	3.1285	0
N3C1W2_CL2_3_4_D	200	5	46	45	46	45	2.17%	3600.00	0	45	3.266	1
N3C1W2_CL2_3_4_E	200	5	46	45	46	45	2.17%	3600.00	0	45	3.2026	1
N3C1W4_CL2_3_4_A	200	5	49	47	49	47	4.08%	3600.00	0	48	4.1599	0
N3C1W4_CL2_3_4_B	200	5	48	47	48	47	2.08%	3600.00	0	47	3.5331	1
N3C1W4_CL2_3_4_C	200	5	48	47	48	47	2.08%	3600.00	0	47	4.1434	1
N3C1W4_CL2_3_4_D	200	5	49	47	49	47	4.08%	3600.00	0	48	4.0631	0
N3C1W4_CL2_3_4_E	200	5	48	46	48	46	4.17%	3600.00	0	46	4.2937	1
N3C2W1_CL2_3_4_A	200	5	37	37	37	37	0.00%	0.00	1	37	2.6921	1
N3C2W1_CL2_3_4_B	200	5	36	35	36	35	2.78%	3600.00	0	35	2.3391	1
N3C2W1_CL2_3_4_C	200	5	36	36	36	36	0.00%	0.00	1	36	1.9651	1
N3C2W1_CL2_3_4_D	200	5	36	35	36	35	2.78%	3600.00	0	36	2.2412	0
N3C2W1_CL2_3_4_E	200	5	36	36	36	36	0.00%	0.00	1	36	1.9591	1
N3C2W2_CL2_3_4_A	200	5	42	41	42	41	2.38%	3600.00	0	42	3.1075	0
N3C2W2_CL2_3_4_B	200	5	41	40	41	40	2.44%	3600.00	0	40	2.5517	1
N3C2W2_CL2_3_4_C	200	5	42	40	42	40	4.76%	3600.00	0	41	3.5799	0
N3C2W2_CL2_3_4_D	200	5	42	41	42	41	2.38%	3600.00	0	42	2.6208	0
N3C2W2_CL2_3_4_E	200	5	43	42	43	42	2.33%	3600.00	0	43	3.8077	0
N3C2W4_CL2_3_4_A	200	5	43	42	43	42	2.33%	3600.00	0	43	3.6288	0
N3C2W4_CL2_3_4_B	200	5	43	42	43	42	2.33%	3600.00	0	42	3.7997	1
N3C2W4_CL2_3_4_C	200	5	46	44	46	44	4.35%	3600.00	0	44	3.1314	1
N3C2W4_CL2_3_4_D	200	5	43	42	43	42	2.33%	3600.00	0	43	3.198	0
N3C2W4_CL2_3_4_E	200	5	43	41	43	41	4.65%	3600.00	0	42	2.9304	0
N3C3W1_CL2_3_4_A	200	5	31	31	31	31	0.00%	0.00	1	31	1.7006	1
N3C3W1_CL2_3_4_B	200	5	33	33	33	33	0.00%	0.00	1	33	1.4951	1
N3C3W1_CL2_3_4_C	200	5	32	32	32	32	0.00%	0.00	1	32	2.167	1
N3C3W1_CL2_3_4_D	200	5	31	30	31	30	3.23%	3600.00	0	30	1.7615	1
N3C3W1_CL2_3_4_E	200	5	32	32	32	32	0.00%	0.00	1	32	1.7708	1
N3C3W2_CL2_3_4_A	200	5	37	37	37	37	0.00%	0.00	1	37	3.184	1
N3C3W2_CL2_3_4_B	200	5	37	36	37	36	2.70%	3600.00	0	37	2.4722	0
N3C3W2_CL2_3_4_C	200	5	37	36	37	36	2.70%	3600.00	0	37	2.2147	0
N3C3W2_CL2_3_4_D	200	5	36	35	36	36	0.00%	150.53	1	36	2.1904	1
N3C3W2_CL2_3_4_E	200	5	36	35	36	36	0.00%	158.27	1	36	2.7112	1
N3C3W4_CL2_3_4_A	200	5	39	38	39	38	2.56%	3600.00	0	39	2.7454	0
N3C3W4_CL2_3_4_B	200	5	39	37	39	37	5.13%	3600.00	0	38	2.9585	0
N3C3W4_CL2_3_4_C	200	5	39	38	39	38	2.56%	3600.00	0	38	2.2148	1
N3C3W4_CL2_3_4_D	200	5	39	37	39	37	5.13%	3600.00	0	38	2.4723	0
N3C3W4_CL2_3_4_E	200	5	38	37	38	37	2.63%	3600.00	0	38	2.8298	0

Table A.16: Detailed Comparison for N200 Class5

Bibliography

[1] M. M. Baldi, T. G. Crainic, G. Perboli, R. Tadei, Branch-and-price and beam search algorithms for the Variable Cost and Size Bin Packing Problem with optional items, Annals of Operations Research, 2014, Volume 222, Issue 1 , pp 125-141

[2] N.Bansal, Z.Liu, A.Sankar, Bin-packing with fragile objects and frequency allocation in cellular networks, Wireless Networks 15,2009, pp. 821-830.

[3] J. O. Berkey, Massively parallel computing applied to the onedimensional bin packing problem, IEEE Transaction(1988) CH2649-2/89/0000/031.

[4] J. O. Berkey, P. Y. Wang, A Systolic-Based Parallel Bin Packing Algorithm. IEEE Trans. Parallel Distrib. Syst. July 1994, Vol. 5, NO. 7 , 769-772.

[5] C. Boutevin, M. Gourgand, S. Norre. Bin packing extensions for solving an industrial line balancing problem. Assembly and Task Planning, 2003. Proceedings of the IEEE International Symposium (2003), pp. 115-121.

[6] W.T.Chan, F.Y.-L.Chin, D.Ye, G.Zhang, Y.Zhang, Online bin packing of fragile objects with application in cellular networks, Journal of Combinatorial Optimization 14, 2007, pp. 427-435.

[7] F. Clautiaux, M. Dell'Amico, M. Iori, A. Khanafer, Lower and upper bounds for the bin packing problem with fragile objects, Discrete Applied Mathematics 163, 2014, pp. 73-86.

[8] M. Dell'Amico, J. C. Daz, M. Iori, The Bin Packing Problem with Precedence Constraints, Operations Research, 2012, Volume 60, Issue 6, pp. 1491-1504

[9] S. Elhedhli, L. Li, M, Gzara, J, Naoum-Sawaya, A Branch-and-Price Algorithm for the Bin Packing Problem with Conflicts, INFORMS Journal on Computing, 2011, Volume 23, Issue 3, pp 404-415

[10] H. Esbensen. A genetic algorithm for macro cell placement. In Proceedings of the conference on European design automation. IEEE Computer Society Press, Los Alamitos, CA, USA, 1992, pp. 52-57.

[11] J. A. George, J. M. George, B. W. Lamar, Packing different-sized circles into a rectangular container, European Journal of Operational Research, August 1995, Volume 84, Issue 3, pp. 693-712.

[12] E, D. Goodman and A. Y. Tetelbaum and Victor M. Kureichik, A Genetic Algorithm Approach to Compaction, Bin Packing, and Nesting Problems, Michigan State University, 1994, 940702

[13] S. Hong, D. Zhang, H. C. Lau, X. Zeng, Y. W. Si, A hybrid heuristic algorithm for the 2D variable-sized bin packing problem, European Journal of Operational Research, October 2014, Volume 238, Issue 1, pp. 95-103

[14] M. Hifi, S. Negre, L. Wu, Hybrid greedy heuristics based on linear programming for the three-dimensional single bin-size bin packing problem, International Transactions in Operational Research, 2014, Volume 21, Issue 1, pp. 59-79

[15] S. Jakobs, On genetic algorithms for the packing of polygons, European Journal of Operational Research, Volume 88, January 1996, Issue 1, pp. 165-181.

[16] R. Karmakar, S. Chattopadhyay, Window-based peak power model and Particle Swarm Optimization guided 3-dimensional bin packing for SoC test scheduling, Integration, the VLSI Journal, June 2015, Volume 50, pp. 61-73

[17] A. Khanafer, F. Clautiaux, E. Talbi, Tree-decomposition based heuristics for the two-dimensional bin packing problem with conflicts, Computers and Operations Research, January 2012, Volume 39, Issue 1, pp. 54-63

[18] S. Martello ,D. Pisinger, D. Vigo, The Three-Dimensional Bin Packing Problem. Operations Research, Volume 48, 2000, Issue 2, pp. 256-267.

[19] M. A. Martinez, F. Clautiaux, M. Dell'Amico, M. Iori, Exact algorithms for the bin packing problem with fragile objects, Discrete Applied Mathematics 10, 2010, pp. 210-223.

[20] M. Quiroz-Castellanos, L. Cruz-Reyes, J. Torres-Jimenez. C. Gmez, H. J. F. Huacuja, and A. C. F. Alvim. A grouping genetic algorithm with controlled gene transmission for the bin packing problem. Computers and Operations Research 2015, 55(Complete) pp. 52-64

[21] R. Sadykov, F. Vanderbeck, Bin packing with conflicts: a generic branch-and-price algorithm, INFORMS Journal on Computing, 2013, Volume 25, Issue 2, pp 244-255