# Landscape Analysis of a Class of NP-Hard Binary Packing Problems

**Khulood Alyahya** K.Alyahya@exeter.ac.uk
Department of Computer Science, University of Exeter, EX4 4QF, UK

**Jonathan E. Rowe** J.E.Rowe@cs.bham.ac.uk
Department of Computer Science, University of Birmingham, B15 2TT, UK

**Abstract**

This paper presents an exploratory landscape analysis of three NP-hard combinatorial optimisation problems: the number partitioning problem, the binary knapsack problem, and the quadratic binary knapsack problem. In the paper, we examine empirically a number of fitness landscape properties of randomly generated instances of these problems. We believe that the studied properties give insight into the structure of the problem landscape and can be representative of the problem difficulty, in particular with respect to local search algorithms. Our work focuses on studying how these properties vary with different values of problem parameters. We also compare these properties across various landscapes that were induced by different penalty functions and different neighbourhood operators. Unlike existing studies of these problems, we study instances generated at random from various distributions. We found a general trend where some of the landscape features in all of the three problems were found to vary between the different distributions. We captured this variation by a single, easy to calculate, parameter and we showed that it has a potentially useful application in guiding the choice of the neighbourhood operator of some local search heuristics.

**Keywords**

Local Search, Random Restarts, Landscape Analysis, Binary Knapsack Problem, Number Partitioning, Quadratic Knapsack, Combinatorial Optimisation Problems, Operator Selection, Plateau, Local Optima, Basin of Attraction, Phase transition, Constraint Optimisation, Penalty Functions, Feasibility Problem.

## 1 Introduction

In the field of evolutionary computation and meta-heuristics, the most common practice of evaluating the performance of a new proposed algorithm is through what is known as "competitive testing", where an extensive number of experiments is performed on benchmark problems and the results are then compared against the performance of other existent algorithms (Burke et al., 2009). Hooker (1995) has warned against this approach more than 20 years ago and argued that it is harmful for research in the field as it gives little or no insight into why or how the algorithm under test is better or worse than the others. Over the past few years, this line of research has been increasingly criticised for not advancing our knowledge and understanding of the algorithm behaviour (Swan et al., 2015; Srensen, 2015). The need for a deep understanding of the problem features and how it relates to algorithm performance has consequently been recognised and encouraged by an ongoing initiative to guide the research in the field towards this direction (ibid.). Indeed, understanding problems features and exploring algorithms weaknesses and strengths has been a popular area of

research both as an independent exploratory landscape analysis of problems (Tayarani-N. and Prügel-Bennett, 2015b,c,a; Prügel-Bennett and Tayarani-N., 2012; Daolio et al., 2015) or as part of automatic algorithm selection and configuration techniques (Smith-Miles and Lopes, 2012; Consoli et al., 2014).

Fitness landscape analysis emerged as an analytical framework to address the need of understanding the relation between problem features and algorithm performance. Indeed, an extensive amount of research has been carried out in the past two decades where various predictive measures were proposed to characterise problem difficulty. However, soon it has been realised that a single general measure that accurately predicts the difficulty of all the problems and that can be computed in polynomial-time cannot exist unless P = NP as rigorously proven by He et al. (2007). These measures, therefore, should be viewed instead as part of a toolbox of techniques to broadly characterise problems. Studying more than one measure or feature can help in getting a broader perspective and increase the chances of capturing various aspects of the problem difficulty.

Three NP-hard binary packing problems are studied in this paper, each of which is an abstraction of many real-world problems. The problems are the number partitioning problem, the binary knapsack problem, and the quadratic binary knapsack problem. We provide a systematic landscape analysis studying a set of properties that we believe to be representative of the problems difficulties and to give insights into the structure of the landscape particularly with respect to local search. We provide a comparative analysis where we compare and contrast the landscape properties across different landscapes that were induced by different neighbourhood operators, penalty functions and problem parameters. To the best of our knowledge, this paper is the first to conduct such extensive and comparative landscape analysis of these problems. The analysis examines landscape properties, neighbourhood operators and differently distributed random instances that have not been studied before in the literature. As a result of the landscape analysis, the study also proposes a good heuristic for determining which local search operator to choose for this class of binary packing problems.

The paper is structured as follows. Section 2 presents preliminaries and definitions. Section 3 introduces, formally, the three problems under study and describes the random generation of their instances. Sections 4 to 6 discuss the properties of the different landscapes of all three problems and examines the similarities and dissimilarities between them. Section 7 examines the performance of local search and relates that to the corresponding landscape properties. The paper is concluded in section 8 with remarks on future work.

## 2 Preliminaries and Definitions

**Search Space** The search space $X$ is the finite set of all the candidate solutions. The fitness functions of all the studied problems in this paper are pseudo-Boolean functions, hence the search space size is $2^n$.

**Neighbourhood** A neighbourhood is a mapping $N : X \rightarrow P(X)$, that associates each solution with a set of candidate solutions, called neighbours, which can be reached by applying the neighbourhood operator once. The set of neighbours of $x$ is called $N(x)$, and $x \notin N(x)$. We consider two different neighbourhood operators: the Hamming 1 operator (*H1*) and the 1+2 Hamming operator (*H1+2*) . The neighbourhood of the *H1* operator is the set of points that are reached by 1-bit flip mutation of the current solution $x$, hence the neighbourhood size is $|N(x)| = n$. The neighbour-

hood of the *H1+2* operator includes the Hamming one neighbours in addition to the Hamming two neighbours of the current solution $x$, which can be reached by 2-bits flip mutation. The neighbourhood size for this operator is $|N(x)| = (n^2+n)/2$. Suppose we lay out the search space in circles around a configuration $x$, so that $x$ is placed in the centre and the configurations that are $h$ Hamming distance away from it lie on the circle of radius $h$ (see figure 1 for an illustrative example when $n = 10$). For a configuration in the $h$-th circle, its *H1* neighbours will be spread out as follows: $h$ of them will reside in the $h-1$ circle, the rest $(n-h)$ will reside in the $h+1$ circle. Its *H1+2* neighbours will be spread out over the $h-2$, $h-1$, $h$, $h+1$, $h+2$ circles as follows: $h(h-1)/2$, $h$, $h(n-h)$, $n-h$, $(n-h)(n-h-1)/2$ respectively.

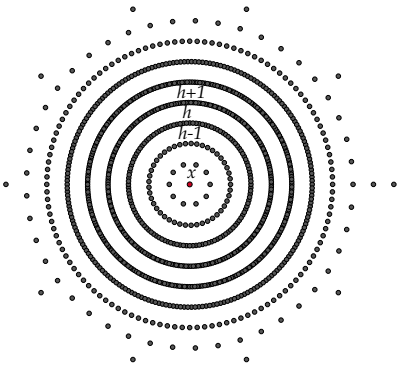



Figure 1: Illustration of the search space layout into circles of radius $h$ around a configuration $x$, points that are $h$-Hamming distance away from $x$ lie on the $h$-th circle.

**Fitness Landscape** The fitness landscape of a combinatorial optimisation problem is a triple $(X, N, f)$, where $f$ is the objective function $f : X \to R$, $X$ is the search space and $N$ is the neighbourhood operator function (Stadler and Stephens, 2002).

**Search Position Type** For a given point $x \in X$ in the landscape, according to the topology and fitness values of its direct neighbourhood, it can belong to one of seven different types of search positions (Hoos and Stützle, 2005). The types are:

- Strict local minimum (SLMIN): $\forall y \in N(x),\ f(y) > f(x)$.
- Non-strict local minimum (NSLMIN): $\forall y \in N(x),\ f(y) \geq f(x)$, and $\exists\ u, z \in N(x)$, such that $f(u) = f(x)$, and $f(z) > f(x)$.
- Interior plateau (IPLAT): $\forall\ y \in N(x),\ f(y) = f(x)$.
- Ledge (LEDGE): $\exists\ u, y, z \in N(x)$, such that $f(u) = f(x), f(y) > f(x)$, and $f(z) < f(x)$.
- Slope (SLOPE): $\forall y \in N(x),\ f(y) \neq f(x)$, and $\exists\ u, z \in N(x)$, such that $f(u) < f(x)$, and $f(z) > f(x)$.
- Non-strict local maximum (NSLMAX): $\forall y \in N(x),\ f(y) \leq f(x)$, and $\exists\ u, z \in N(x)$, such that $f(u) = f(x)$, and $f(z) < f(x)$.
- Strict local maximum (SLMAX): $\forall y \in N(x),\ f(y) < f(x)$.

For the purpose of this paper, we use the term *local optimum* to refer to both strict and non-strict local optimum.

**Global Optima** Assuming maximisation, a point $x \in X$ is a *strict global maximum* if it is a strict local maximum and $\forall y \in X, f(x) \geq f(y)$, and a point $x \in X$ is a *non-strict global maximum* if it is a non-strict local maximum and $\forall y \in X, f(x) \geq f(y)$.

**Plateaux** A plateau is a set of connected non-strict local maxima, with or without interior plateau points. An exit is a neighbour of one or more configurations in the plateau, which shares the same fitness value of the plateau, but has an improving move. An exit could be a non-strict local minimum (maximum when minimising) or a ledge. We call a plateau open when it has at least one exit, otherwise we call it closed. We call a plateau of non-strict global maximum, a global plateau. Obviously, all global plateaux are closed. We illustrated our definitions in Figure 2. Collecting information about the different plateaux types gives an insight into the different plateaux regions in the problems and can help inform the algorithm design and the choice of search operators. For example, a problem with mostly open than closed plateaux motivates the use of plateaux moves.

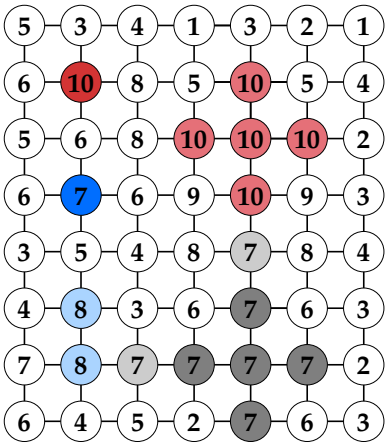



Figure 2: Schematic illustration (following Prügel-Bennett and Tayarani-N. (2012)) of our definitions of strict global optima, strict local optima, and global, closed and open plateaux. Two points are neighbours if there is an edge between them. Assuming maximisation: a strict global optimum is shown by the single dark red point of fitness 10; A strict local optimum is shown by the single dark blue point of fitness 7; A global plateau is shown by the light red region of size 5 and fitness 10; A closed plateau is shown by the light blue region of size 2 and fitness 8; and an open plateau is shown by the grey region of size 5 (7 when considering the exits) and fitness 7, the open plateau has two exits (light grey), one (a non-strict local minimum) to the global plateau and one (a ledge) to the closed plateau.

**Local Search** The local search algorithm under study in this paper is the steepest ascent(descent when minimising) with no plateau moves and with random restarts.

**Basin of Attraction** The attraction basin $B(x^*)$ of an optimum $x^* \in X$ is the set of points that leads to it after applying local search to them, $B(x^*) = \{x \in X \mid \text{localsearch}(x) = x^*\}$. The basin of a plateau is the union of the basins of its configurations. The neighbours of a point $x$ are evaluated in order from left to right, with respect to bit flips, in the case of having more than one neighbour with the

best improving move, the first one is always selected. Of course, this deterministic way of choosing the improving move could introduce some bias to the size of the basin. However, there was only a small subset of such configurations in the landscapes of the instances we have studied. Thus, we speculate that the bias, if any, will be quite small. A possible way to break the tie and avoid the bias is to choose randomly between the best improving configurations. This method, however, will render the landscape structure stochastic. To study the attraction basin shape, we use the return probability concept from (Prügel-Bennett and Tayarani-N., 2012). The return probability to an optimum starting from a Hamming sphere of radius $h$ around it is given by $p_r(h)$. We randomly sample $s$ configurations that are $h$ Hamming distance away from an optimum, we then apply local search to them and calculate the fraction that led to the starting optimum. The sample size $s$ is obtained from $s_0 = \frac{z^2_{\alpha/2}\hat{p}(1-\hat{p})}{e^2}$ and $s_1 = \frac{s_0 S}{s_0+(S-1)}$ (Alyahya and Rowe, 2016), by setting the estimated proportion $\hat{p} = 0.5$, the error margin $e = 0.005$, and the z-score $z_{\alpha/2} = 1.645$ (corresponding to $90\%$ confidence level), $S$ is the number of configurations in the Hamming sphere $h$.

## 3 Problems

We study three NP-hard problems: the number partitioning problem (NPP), the binary knapsack problem (0-1KP), and the quadratic binary knapsack problem (0-1QKP). All of the three problems are similar in nature, in the sense that all of them fall into a class of NP-hard binary packing problems related to the 0-1 knapsack problem. A special case of the 0-1KP known as the subset sum problem is a generalisation of the NPP. The 0-1QKP is a variant of the 0-1KP where the profit of an item depends also on the other selected items. The 0-1KP and the 0-1QKP are constrained optimisation problems and their search space is partitioned into a feasible and an infeasible region.

### 3.1 Number Partitioning Problem

The NPP is a classical problem in theoretical computer science and one of Garey and Johnson's six basic NP-complete problems (Garey and Johnson, 1979). Given a set $W = \{w_1, \ldots, w_n\}$ of $m$-bit positive integers (weights) drawn at random from the set $\{1, 2, \ldots, M\}$ with $M = 2^m$, the goal is to partition $W$ into two disjoint subsets $S, S'$ such that the discrepancy between them $|\sum_{w_i \in S} w_i - \sum_{w_i \in S'} w_i|$ is minimised. A partition is called perfect, if the discrepancy between the two subsets is 0 when the sum of the original set is even, or 1 when the sum is odd. Equivalently, the problem can be viewed as minimising: $\max\left\{\sum_{w_i \in S} w_i,\ \sum_{w_i \in S'} w_i\right\}$, the maximum sum over the two subsets. Let $x \in \{0,1\}^n$, the objective function to be minimised can be defined as:

$$f(x) = \left| \sum_{i=1}^{n} w_i x_i - \sum_{i=1}^{n} w_i (1 - x_i) \right| \tag{1}$$

The binary representation of NPP creates a symmetry in the search space, in the sense that a solution and its bitwise complement have the same fitness value. Thus, the number of unique solutions is $\leq 2^{n-1}$. The NPP is NP-hard in the weak sense (Garey and Johnson, 1979), that is, there exists an algorithm that can solve it in pseudo-polynomial time through dynamic programming. The complexity of such an algorithm, $\mathcal{O}(n2^{\log_2 \sum_{i=1}^{n} w_i})$, is polynomial in the number of weights and the sum of the weights but exponential in the number of bits required to represent the sum. As Garey

and Johnson (Garey and Johnson, 1979) note, such an algorithm will display an exponential behaviour only when extremely large input numbers are allowed. The running time of such an algorithm would thus exhibit an exponential behaviour as $M$ grows large.

NPP undergoes a sudden phase transition from solvability (a perfect partition exist) to insolvability (a perfect partition doesn't exist), determined by the control parameter $k = \log_2(M)/n$, which corresponds to the number of the bits required to encode the numbers in the set divided by the size of the set. For $\log_2(M)$ and $n$ tending to infinity, the transition occurs at the critical value of $k_c = 1$, such that for $k < 1$, there are many perfect partitions with probability tending to 1, whereas for $k > 1$, the number of perfect partitions drops to zero with probability tending to 1 (Borgs et al., 2001). A more detailed parameterisation of the critical value of the control parameter is given by the following [1] (Mertens, 2001): $k_c = 1 - \frac{\ln(\frac{\pi}{6}n)}{2n\ln(2)}$.

The transition between the two phases appears in the size of the problem backbone. The backbone notion is borrowed from statistical physics and has been generalised to optimisation problems by Slaney and Walsh (2001), where it is defined as the set of decisions with fixed outcomes in all optimal solutions. In NPP, the pairs of weights that are placed in the same subset or in opposite subsets in all optimal solutions of an instance form the backbone of that instance. There is a very sharp increase in the backbone size of the optimal solutions in the NPP as one approaches the phase transition boundary, after which the backbone tends to be complete giving a unique optimal solution (Slaney and Walsh, 2001). In the literature, the effect of this phase transition has been shown in the computational complexity of some exact solvers such as the complete Karmarkar-Karp differencing algorithm (Mertens, 2001). Where instances with $k < k_c$ were "easy-to-solve" and the ones with $k > k_c$ were "hard-to-solve".

### 3.2 0-1 Knapsack Problem

Given a knapsack of capacity $C$ and a set of $n$ items each with associated weight $w_i$ and profit $p_i$, the aim is to find a subset of items that maximises

$$f(x) = \sum_{i=1}^{n} x_i p_i \tag{2}$$

subject to:

$$\sum_{i=1}^{n} x_i w_i \leq C,\ x \in \{0,1\}^n \tag{3}$$

where

$$C = \lambda \sum_{i=1}^{n} w_i,\ 0 \leq \lambda \leq 1 \tag{4}$$

The binary vector $x = (x_1, \ldots, x_n)$ represents the decision variable where $x_i = 1$ when item $i$ belongs to the subset and $x_i = 0$ otherwise. We study instances where $p_i$ and $w_i$ are positive integers drawn from the set $\{1, 2, \ldots, M\}$. Like NPP, the 0-1KP is NP-hard in the weak sense (Garey and Johnson, 1979).

Note that the 0-1KP search space, $X = \{0,1\}^n$, is partitioned into a feasible region $F = \{x \in X \mid \sum_{i=1}^{n} x_i w_i \leq C\}$ and an infeasible region $INF = X \setminus F$. For $\lambda = 1$ there

[1] A more rigorous derivation of the transition point can be found in (Borgs et al., 2001).

are no infeasible solutions and as the value of $\lambda$ decreases the size of the infeasible region increases until $INF = X$ when $\lambda = 0$. We define the boundary between feasible and infeasible regions as the set of feasible configurations that have at least one infeasible neighbour, $B = \{x \in X \mid x \in F \wedge \exists y : (y \in N(x) \wedge y \in INF)\}$. We mostly study instance with moderate constraint ($\lambda = 0.5$), such instances have the largest boundary sizes and thus the largest number of optima.

Randomly generated instances of the 0-1 KP can be classified into different types based on the relation between the item's profit and weight. We study 11 types, which have been the focus of several studies in the literature, each with different properties that could influence the performance of problem solvers (Pisinger, 1999, 2005; Martello et al., 1999; Caccetta and Kulanoot, 2001). The instance types are: uncorrelated, weakly correlated, strongly correlated, inverse strongly correlated, subset sum, uncorrelated spanner,weakly correlated spanner, strongly correlated spanner, multiple strongly correlated, profit ceiling, and circle. The definition of the instance types are taken from (Pisinger, 2005) [2].

The problem type, subset sum, has a similar phase transition to NPP determined by the same control parameter $k = log_2 M/n$ (Sasamoto, 2003). The parametrisation of the critical value $k_c$ for subset sum is given in (Sasamoto et al., 2001).

### 3.2.1 Constraint handling

We use a penalty function as a constraint handling method. An infeasible solution $x$ that violates the given constraint is penalised by a value $\mathrm{Pen}(x) > 0$, $\mathrm{Pen}(x) = 0$ for a feasible solution $x$. The objective functions after adding the penalty term is $f(x) = \sum_{i=1}^{n} x_i p_i - \mathrm{Pen}(x)$. The choice of an appropriate penalty function is very critical for inducing smoother landscapes and guiding the search process to good feasible regions. Careful design of appropriate penalty functions is particularly important for this class of problems since local optima in all covering and packing problems lie in the boundary of the feasible region (Gottlieb, 2001). Assigning a lower fitness value to infeasible solutions than all feasible solutions is hence of critical importance for successful penalty-based search. Some penalty-based algorithms for the 0-1KP suffer from the feasibility problem, that is, they often terminate with completely infeasible solutions due to inappropriate choice of the penalty function (Olsen, 1994).

Various penalty functions have been proposed for the 0-1KP (Olsen, 1994; Michalewicz and Arabas, 1994) and its generalisation the multiple knapsack problem (Gottlieb, 2001; Khuri et al., 1994). We studied three penalty functions proposed in (Michalewicz and Arabas, 1994), which differ in the growth of the penalty value with respect to the degree of constraint violation, namely: logarithmic, linear and quadratic. We also add the term $\sum_{i=1}^{n} p_i$ to the penalty function as an offset term that insures that all infeasible solutions achieve lower fitness values than all feasible solutions (Gottlieb, 2001). Otherwise, in the logarithmic case for instance, the entire search space becomes part of the all ones solution $x = (1, \cdots, 1)$ basin. The penalty functions, in order from weak to strong, are as follows:

$$\mathrm{Pen}(x) = \log_2 \left(1 + \rho \left(\sum_{i=1}^{n} x_i w_i - C\right)\right) + \sum_{i=1}^{n} p_i \tag{5}$$

$$\mathrm{Pen}(x) = \rho \left(\sum_{i=1}^{n} x_i w_i - C\right) + \sum_{i=1}^{n} p_i \tag{6}$$

[2] See the supplementary material for a detailed description of our process of generating the instances.

$$\text{Pen}(x) = \left( \rho \left( \sum_{i=1}^{n} x_i w_i - C \right) \right)^2 + \sum_{i=1}^{n} p_i, \tag{7}$$

where $\rho = \max_{i=1,\ldots,n} \{p_i\} / \min_{i=1,\ldots,n} \{w_i\}$. We found some evidence that the linear penalty function appears to be the best choice, in terms of local search performance and correlation of basin size and fitness, out of the three investigated functions. The logarithmic penalty function is a poor choice since it creates a strict local optimum, the all ones solution $x = (1, \cdots, 1)$, in the infeasible region. While both the linear and quadratic functions do not create any local optima in the infeasible region, the strong penalty enforced by the quadratic function seems to direct the infeasible configurations to be part of the basins of optima with lower quality, as opposed to the linear penalty function. In the rest of this paper, we use the linear penalty function (eq. 6) with all instance types except for the subset sum. Applying this penalty function to infeasible solutions in a subset sum instance assigns equal fitness values for all infeasible solution, thus creating large plateaux in the landscape. Instead, we simply set the fitness of an infeasible solution in a subset sum instance to the negative of the amount it exceeded the knapsack capacity by, $f(x) = C - \sum_{i=1}^{n} x_i w_i$.

### 3.3 Quadratic 0-1 Knapsack Problem

Given a knapsack of capacity $C$ and a set of $n$ items each with associated weight $w_i$, and profits according to an $n \times n$ non-negative integer matrix $P = p_{ij}$, where $p_{jj}$ is the profit achieved if item $j$ is selected and $p_{ij} + p_{ji}$ is the profit achieved if both items $i$ and $j$ are selected (for $i < j$) (Gallo et al., 1980). The aim of the 0-1QKP is to find a subset of items that maximises the profit without exceeding the knapsack capacity. The density of the profit matrix, that is the percentage of non-zero elements, is given by $\Delta$. The quadratic objective function to be maximised is as follows:

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} p_{ij} x_i x_j \tag{8}$$

subject to the linear constraint

$$\sum_{i=1}^{n} w_i x_i \leq C, \; x \in \{0,1\}^n \tag{9}$$

where

$$C = \lambda \sum_{i=1}^{n} w_i, \; 0 \leq \lambda \leq 1. \tag{10}$$

The 0-1QKP is NP-hard in the strong sense (Garey and Johnson, 1979; Pisinger, 2007), it cannot be solved by a pseudo-polynomial time algorithm unless P=NP (Garey and Johnson, 1979). We only consider instances where the profit matrix is symmetric, i.e. $p_{ij} = p_{ji}$. We study instances where $p_{ij}$ and $w_i$ are positive integers drawn at random from the set $\{1, 2, \ldots, M\}$. We only study instances where the profits and the weights are uncorrelated. As with the 0-1KP and for the same motivation we study instances where $\lambda$ is set to 0.5.

### 3.3.1 Constraint handling

As with the 0-1KP, we want to allow the infeasible solutions to be part of the searchable space and we want to penalise them proportional to the degree of violation of the constraint. Also, we want all infeasible solutions to have lower fitness values than all the feasible solutions. To ensure that, we added the offset term $\sum_{i=1}^{n}\sum_{j=1}^{n} p_{ij}$ to the penalty function. We ruled out the logarithmic penalty function as it creates a strict local optimum in the infeasible region as in the 0-1KP case. We also ruled out the use of the linear one as it was found to create a strict local optimum (the all ones solution $x = (1, \cdots, 1)$) in the infeasible region of some instances with highly dense profit matrix. It was also found to create some open and closed plateaux in the infeasible region of some instances with various values of $\Delta$. The quadratic penalty function was found to induce a landscape with a smooth infeasible region that does not have any local optima or plateaux. Therefore, we use the quadratic penalty function to handle the constraint in this problem. The function is defined as follows:

$$\mathrm{Pen}(x) = \left(\rho\left(\sum_{i=1}^{n} x_i w_i - C\right)\right)^2 + \sum_{i=1}^{n}\sum_{j=1}^{n} p_{ij}, \tag{11}$$

where $\rho = \max_{i,j=1,\ldots,n}\{p_{ii} + p_{ij} + p_{ji}\} / \min_{i=1,\ldots,n}\{w_i\}$.

### 3.4 Random Instance Generation

Most of the existing studies of these problems only consider instances where weights are drawn at random from a uniform distribution. We are interested in studying if and how the landscape properties of instances generated randomly from different distributions vary. Thus, we generate instances with weights drawn randomly from five different discrete probability distributions: uniform, normal, negatively skewed, positively skewed and bimodal distribution with peaks at both ends. We used arbitrary-precision arithmetic for large ranges that exceed the fixed precision range. We investigate setting the phase transition control parameter $k$ to 0.4 and 1 in all the studied problems and additionally to $1.2$ in NPP. For the 0-1QKP we explore the effect of varying the profit matrix density on the landscape by studying instances with $\Delta = 0.1,\ 0.25,\ 0.5,\ 0.75,\ 0.95,$ and $1$. We study instances of size $n = 14, 16, 18, 20, 22, 30, 100$. For each parameter combination, landscape properties are collected from exhaustively enumerated search space of 600 instances for $n < 30$ and sampled from 500 instances for $n \geq 30$ .

## 4 Search Position Types

The common finding across all of three problems is that no configuration of type IPLAT was found in either of the *H1* or *H1+2* landscapes across the different distributions of the weights, the different values of $k$, and for all $\Delta > 0.25$ in the 0-1QKP. For a sparse 0-1QKP profit matrix $\Delta \leq 0.25$, very few configurations of type IPLAT were found, which is not surprising, since the very low density of the profit matrix results in many solutions sharing similar objective values. The absence of IPLATs and NSLMAXs in NPP is an implication of its elementary landscape.The NPP has an elementary landscape under the *H1* operator when the objective function is the square of the discrepancy(Grover, 1992). This has an implication on the types of plateaux and search positions in its landscapes (Whitley et al., 2008). The first implication is that configurations of type IPLAT can only exist when the entire landscape is flat. The second implication is that exits of open plateaux can only be LEDGEs not NSLMAXs. On both

landscapes of NPP, there are always two configurations of type SLMAX: the all zeros solution $x = (0, \cdots, 0)$ and its bitwise complement. In the infeasible region of 0-1KP , all the found configurations were of the following types: LEDGE, SLOPE, SLMIN and NSLMIN. No strict optima or plateaux were found in the infeasible region. In the *H1* landscape of the infeasible region, the configurations were mainly of type LEDGE with both values of $k$. In the *H1+2* landscape of the infeasible region, the configurations were mainly of types LEDGE and SLOPE when $k = 0.4$, and mainly LEDGE when $k = 1$. In the feasible region of the *H1+2* landscape, when $k = 0.4$, the found configurations were mainly of types LEDGE, SLOPE , NSLMAX and SLMAX in all the problem types. Apart from the uncorrelated and uncorrelated spanner instances, where no configurations of type NSLMAX or SLOPE were found. When $k = 1$, the SLOPE and NSLMAX types disappear, aside from very few configurations in normal and negatively skewed instances of the profit ceiling and circle problem types. The feasible region of the *H1* landscape is similar to its infeasible region, in that, no apparent changes were observed between the values of $k$. In general, the configurations were of type SLMAX and LEDGE in this region. The configurations in the infeasible region of 0-1QKP were of types: SLMIN, LEDGE, and SLOPE. The SLOPE configurations seem to disappear in the infeasible region of the *H1+2* landscape when $k$ goes from 0.4 to 1. Apart from that, not much difference is found between the two values of $k$ across all the different $\Delta$ and for both landscapes and regions.

## 5 Properties of Optima and Plateaux

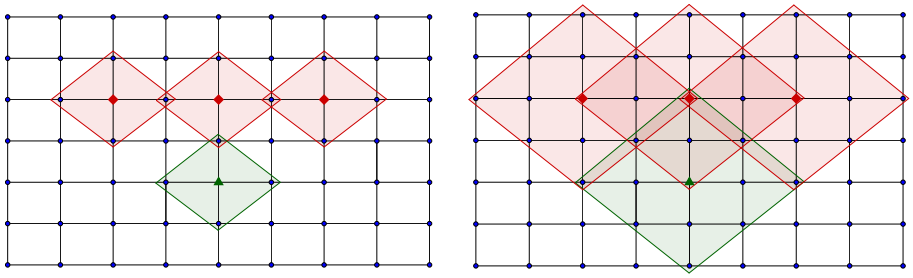
### 5.1 Number of optima and plateaux

In NPP, the number of global optima is highest when $k = 0.4$ and it starts to decrease as we approach the critical phase transition point and keeps decreasing as we cross the point until we have only two optimal solutions. Similar behaviour has been observed for the number of non-strict local optima, where it starts to decrease as we approach the phase transition until it reaches zero in the phase beyond the critical point. That is true for both the *H1* and *H1+2* landscapes and for all the different distributions. The results can be found in our paper (Alyahya and Rowe, 2014). In the 0-1QKP and in all problem types of the 0-1KP apart subset sum, the number of global optima is not effected by the different values of $k$. In general there was only one global optimum apart from few exceptions with slightly higher number in very sparse 0-1QKP profit matrix ($\Delta = 0.1$) and in strongly correlated, multiple strongly correlated, and profit ceiling problems. In subset sum, similar to NPP, the optimal solutions number is highest (around $10^3$ for $n = 20$) when $k = 0.4$, and it drops down to only one when $k = 1$.

The number of strict local optima does not change very much between the different values of $k$ regardless of the distribution from which the weights are chosen and regardless of the used operator. This is true for all the three problems. In the 0-1KP and in the 0-1QKP, the variation in the number of local optima in this landscape between the different problem types and between the different values of $\Delta$ was found to be very small (with the exception of slightly higher number in instances with very sparse profit matrix $\Delta = 0.1$). The number of strict local optima was found however to vary largely between distributions in the *H1* landscape in all problems. In particular, the negatively skewed and the normal distributions have the largest number of strict local optima (between 8-15% of the search space) while the positively skewed and the two peaks distributions have the fewest (around 1% of the search space). The number of strict local optima drops in the *H1+2* landscape compared to the *H1*, which is expected, however the difference between the two landscapes is large. The difference is

in two orders of magnitude in the negatively skewed and the normal distributions and in one orders of magnitude for the rest of the distributions. In the 0-1QKP and in the uncorrelated, weakly correlated, uncorrelated and weakly correlated spanner instances of 0-1KP the difference is around 4 orders of magnitude. The number of local optima in the 0-1QKP and uncorrelated 0-1KP is very low in this landscape with medians $< 10$ for $n = 20$.

We believe that the underlying relation between the number of strict local optima in the *H1* landscape and the different distributions can be explained by the variability of the weights. To capture this with a single parameter, that does not require the knowledge of the underlying distribution of the weights, we propose using the coefficient of variation ($CV$) of the weights, which provides a measure of relative variability or dispersion. Roughly speaking (see Figure 4), normal and negatively skewed instances map to the region $\leq 0.3$, uniform instances map to the region between $0.4$ to $0.7$, positively skewed and two peaks instances map to the region $> 0.8$. The density of the region $\leq 0.3$ is higher because both the normal and the negatively skewed map to this small region. The $CV$ seems to capture most of the variation in the number of strict local optima in the *H1* landscape as the two correlate very strongly and negatively across the different values of $n$ and $k$ we studied. To explain the intuition behind this strong correlation, consider an example of two extreme cases of NPP, the first is when all the weights are the same (small $CV$) and the second case is when one weight is equal to the sum of the rest of the weights (large $CV$). In the first case there are $\binom{2n}{n} \sim \frac{4^n}{\sqrt{\pi n}}$ possible ways to split the weights into two subsets, while in the second case there are only two.



(a) Small neighbourhood (b) Large neighbourhood

Figure 3: A schematic illustration showing how applying a larger neighbourhood operator as opposed to a smaller one, can reduce the number of optima, but can also introduce plateaux. (a) Assuming minimisation and under the small neighbourhood operator: the red diamond-shaped nodes are strict optima with the same fitness value, and the green triangle-shaped node is a strict optimum with a higher fitness value than the red optima. The rest of the nodes are either a local maximum, a slope or a ledge at a higher fitness value than the four optima. The shaded areas indicate the neighbourhood of the optimum at its centre. (b) After applying the larger neighbourhood operator, the green optimum is no longer an optimum but a slope. The group of red optima however have now formed a closed plateau.

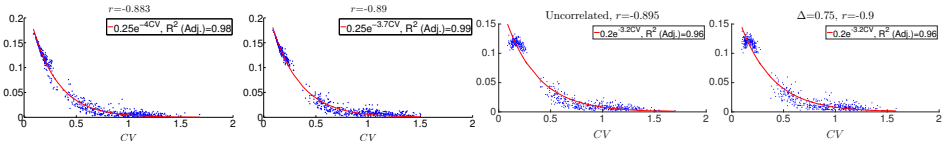
In the phase below the phase transition point where plateaux exists in the NPP, the majority of the plateaux in the *H1* landscape were open plateaux with very few global ones. The number of configurations in each plateau is very small, for example when $n = 20$ the size of the global and closed plateaux was found to be only 2 to 3 non-strict

local optima (remember all the plateaux are composed of NSLMIN only). All the open plateaux we found are composed of only one non-strict local optimum and the majority of them have only one exit. There are more global plateaux in the *H1+2* landscape, but fewer number of open and closed plateaux. The number of exits from open plateaux is larger in this landscape as expected. The size of all the plateaux in this landscape is also larger than that in the *H1* landscape, it can reach up to 300 for global plateaux and around 50 for open and closed ones. In the 0-1KP, plateaux were only found in the *H1+2* landscape and mainly when $k = 0.4$. For most of the problem types, the majority of the found plateaux were closed plateaux. For the subset sum instances, the majority of plateaux were global and closed plateaux. Most of the plateaux found in the profit ceiling instances were open and closed plateaux. In all the problem types, most of the found plateaux have very small sizes, around two or three configurations. The largest found plateaux were less than 10 configurations in all $n = 20$ instances, apart from the profit ceiling and circle instances where the largest found plateaux were composed of around 30 configurations. However, these large plateaux were rarely found. The number of exits in open plateaux was found to be also quite small in all the problem types (mainly between 1-3 exits). In the 0-1QKP, similar results have been found, in that the closed and open plateaux were of very small size, with exception of relatively large number of exits in open plateaux in very sparse profit matrix. As the density ($\Delta$ ) of the profit matrix increases, the plateaux decrease until they disappear in very dense profit matrix instances. The observation that the *H1+2* landscape have sometimes more plateaux than the *H1* may seem counter-intuitive at first, since applying a larger neighbourhood operator as opposed to a smaller one, usually has the positive effect of reducing the number of optima and plateaux. However, it can also have an effect of introducing new larger plateaux. A schematic illustration of this mechanism is shown in figure 3. The figure shows how after applying the larger neighbourhood operator, the same-fitness strict optima became connected forming a closed plateau. If the green triangle-shaped optimum had a better fitness value than the red diamond-shaped optima, then the figure shows how two open plateaux, sharing the same exit and each of size one, can be formed.

### 5.2 Average number of strict local optima

The average proportion of the strict local optima in the *H1* landscape when the NPP weights are drawn from uniform distribution follows this formula $\sqrt{\frac{24}{\pi}}n^{-3/2}$, derived using statistical mechanics (Ferreira and Fontanari, 1998). Based on the strong and negative correlation between the $CV$ and the strict local optima number in all the three problems, we propose a generalised formula for estimating the average proportion of strict local optima in the *H1* landscape, which depends only on the $CV$ of the weights and the size of the problem: $a\, e^{-b\, CV}$, where the values of the coefficients $a$ and $b$ depend on the problem and on $n$. Figure 4 shows the estimation of the fraction of the strict local optima using this formula. The values of $a$ and $b$ were determined by least-squares regressions. The proposed formula seems to be a good approximate fit of the average number of strict optima in all the problem sizes we studied, in particular for the NPP. In the 0-1KP and 0-1QKP, it seems to be noisier around small $CV$ values ($\leq 0.3$).

In the 0-1KP, the correlation between the $CV$ and the number of strict local optima in the *H1* landscape is strong and negative apart from highly constrained instances ($\lambda \leq 0.2$) and weakly constraint instances ($\lambda > 0.8$) where the correlation is strong and positive. In small $CV$ instances, most of the solutions are infeasible when highly constrained and feasible when weakly constrained. This consequently decreases the




(a) NPP $k = 0.4$ (b) NPP $k = 1.2$ (c) 0-1KP (d) 0-1QKP

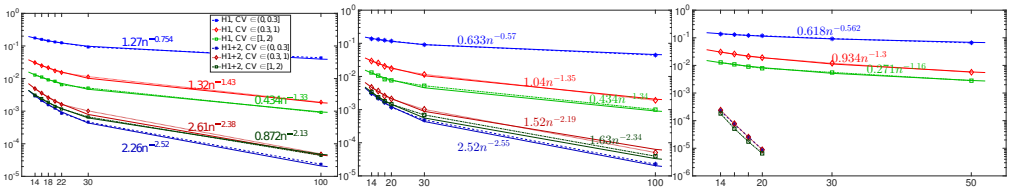
Figure 4: The fraction of strict local optima in the *H1* landscape versus CV. The results are for 600 instances of size $n = 20$. $k = 1$ for 0-1KP and 0-1QKP. Pearson's correlation coefficient $r$ between the two quantities is shown for each plot.

boundary size and thus the number of strict local optima. On the other hand, the larger weights in large $CV$ instances, prevent many solutions from becoming feasible when weakly constrained and allow many solutions to be feasible when highly constrained. This makes the boundary size in large $CV$ instances relatively larger in both cases, causing the optima number to be higher than that in small $CV$ instances. In fact, when highly constrained, the mechanism of solving large $CV$ instances becomes similar to that of solving small $CV$ instances, in that the problem becomes about fitting the small similar weights into the knapsack.

To easily study the growth behaviour of the number of strict local optima as the problem size increases, we grouped the instances based on their $CV$ values into three intervals: $(0, 0.3]$, $(0.3, 1)$, and $[1, 2)$. Figure 5 shows the decay of the proportion of number of strict local optima as $n$ increases. The results for $n = 30, 100$ are the SRS (Alyahya and Rowe, 2016) estimates obtained with the sample sizes $s = 10^5, 5 \times 10^5$ respectively. All the proportions seems to decrease polynomially with $n$ in the form $an^{-b}$. The largest decay happens in the landscape of *H1+2* and the smallest in the *H1* landscape of the interval $(0, 0.3]$. The proportion of the strict local optima appears to decay faster in the *H1+2* landscape compared to the *H1* across all the intervals. The proportion in the *H1+2* landscape of 0-1QKP and the following 0-1KP types: uncorrelated, weakly correlated, uncorrelated spanner, weakly correlated spanner, strongly correlated spanner, and multiple strongly correlated, seems to be smaller than what the SRS with the above sample sizes can detect. This was evident by the negative lower bound of the 95% $\mathrm{CI}_{\mathrm{AC}}$ of the obtained estimates indicating that the point estimates are greatly overestimating the real proportions. Therefore, we did not include these estimates in figure. We also did not fit the decay of the proportions with the form $an^{-b}$, since we are only left with four close data points. We are unable to comment on the growth of the number of strict local optima in the *H1+2* landscapes of these instances. However, the growth of the strict local optima in the *H1+2* landscape of all the remaining problems and the growth in all the *H1* landscapes seems to be exponential with $n$.

### 5.3 Quality of optima and plateaux

The difference in the number of optima between the two landscapes is no doubt an important feature, but another equally important one is the difference in quality of optima between the two. Obviously, the quality of the optima in the *H1+2* landscape is at least as good or better than that in the *H1* as every optimum in the *H1+2* landscape is also an optimum in the *H1*. However, we want to examine how the difference in quality between the two landscape changes across the $CV$ values. To obtain a measure of quality that is independent of the problem instance and that does not require the




(a) NPP (b) 0-1KP, Profit ceiling (c) 0-1 QKP, $\Delta$ =0.75

Figure 5: The decay of the strict local optima proportion as $n$ grows ($k = 1$). The results are averaged over 600 instances for each $n \leq 22$ and over 500 instances for $n = 30, 100$. The number of strict optima is estimated for $n = 30, 100$ using SRS, the sample sizes are $s = 10^5, 5 \times 10^5$ respectively. The solid lines in (b) were obtained using least-squares fit.

knowledge the optimal solution, we measure the quality of an optimum $x$ in a given instance as $f(x)/\sum_{i=1}^{n} w_i$ in NPP, $f(x)/\sum_{i=1}^{n} p_i$ in 0-1KP, and $f(x)/\sum_{i=1}^{n}\sum_{j=1}^{n} p_{ij}$ in 0-1QKP. Figure 6 shows selective results of the three problems [3]. Note that the distributions in the figure cannot be used directly to infer the quality of the local optima relative to the global since the distributions are calculated over multiple instances. In NPP, the number of optima with relatively poor quality in the *H1* landscape of instances with very small $CV$ is very high. The number starts to decrease as the $CV$ values increases. This behaviour is consistent across all $k$ values and all $n$. In the 0-1KP, there is a clear difference among the various problem types in terms of the overall quality of the optima and in the difference between the quality of the optima in the two landscapes. For example, in the uncorrelated and the uncorrelated spanners instances there is a large difference in the quality of optima between the two landscapes and that difference does not seem to change much across the $CV$ intervals. On the other hand, the quality of the optima in the two landscapes seems to be similar in the circle instances and that similarity seems to increase as the $CV$ value increases. The case in the subset sum instances is perhaps the most similar to that in the NPP, in that the quality of the optima in the *H1* landscape gets better as the $CV$ value increases, which in turn decreases the difference in the quality between the two landscapes. In general, and apart from the uncorrelated and the uncorrelated spanner instances, the difference in the quality of optima between the two landscapes appears to decrease as the $CV$ value increases. The 0-1QKP is similar to the uncorrelated problem types in the 0-1KP, across all values of $\Delta$, the quality of optima in the *H1+2* landscape is better than that in the *H1*, and this difference in the quality does not seem to change much across the $CV$ values.

## 6 Basins of Attraction

### 6.1 Basin size

In the NPP, as with the number of strict optima, the basin sizes do not seem to change much across different values of $k$ for all the different distributions and for both landscapes. In all the problems, the size of the basins in the *H1+2* landscape is as expected larger than that in the *H1*. For both landscapes, the distribution of the basin sizes in all the instances we studied was found to be highly skewed to the right, with many small basins and only few large ones. Similar skewness in the distribution of basin sizes was reported in other combinatorial problems, for example in the flow-shop scheduling

[3] See supplementary material for the full results.

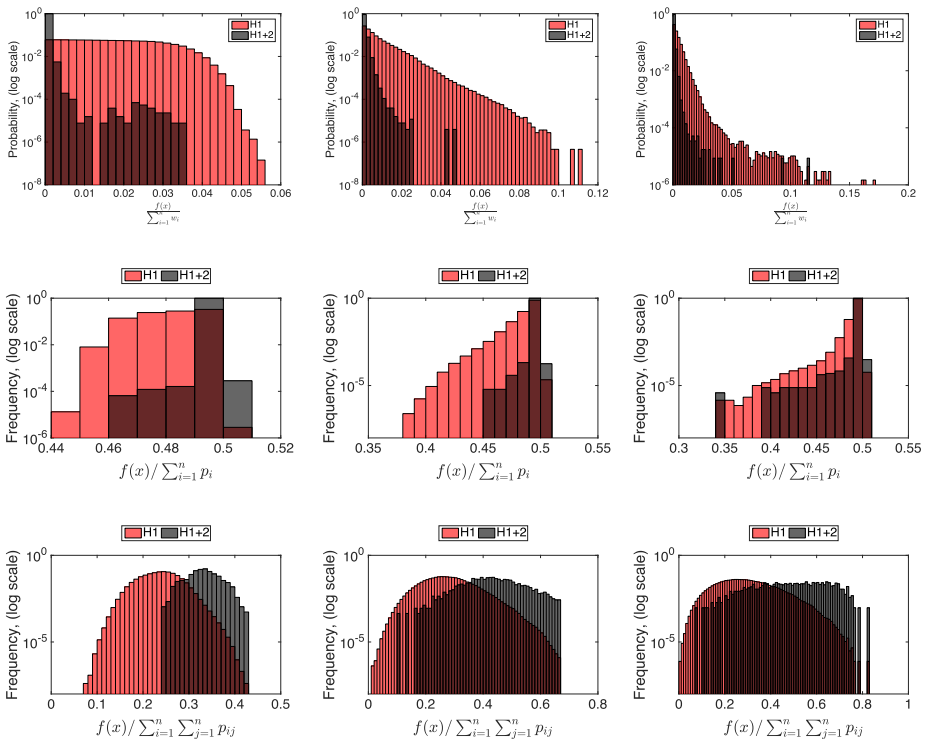


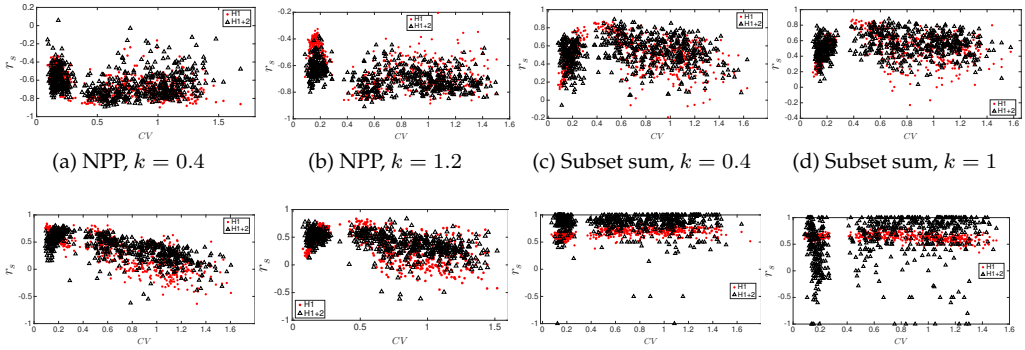

Figure 6: The quality of optima and plateaux in the *H1* and *H1+2* landscapes across the different values of $CV$: $0 < CV \leq 0.3$ (left), $0.3 < CV < 1$ (middle), $1 \leq CV < 2$ (right). The rows show respectively, starting from the top, the results for $n = 20$ and $k = 1$ of NPP, 0-1KP profit ceiling, and 0-1QKP with $\Delta$ =0.75. The data in each plot includes all optima and plateaux found in 600 instances of the respective problem.

problem where the log-normal distribution was found to be a plausible model of the basin size (Reeves and Eremeev, 2004). The basin sizes in the *H1* landscape of all the problems increases with the $CV$ until their sizes become similar to those of the *H1+2* landscape. In the *H1* landscape of NPP, 20% of the basins cover around 60-70% of the search space in instances with $CV \leq 0.3$, while they cover around 35-50% of the search space in instances with $CV > 1$. This discrepancy between the two $CV$ intervals seems to continue to exist in the *H1+2* landscape, though at a lower degree. In the 0-1KP, the largest basins in the uncorrelated problem type quickly covers large area of the search space in the *H1* landscape, where around 60% to 90% of the search space is covered by only 20% of the basins. The same percentage of the basins covers around 40% to 70% of the search space in the weakly correlated problem type, and around 40% to 60% in the subset sum problem type. For the *H1+2* landscape. The results of the uncorrelated spanner are similar to the results of the uncorrelated problem type. The results of the weakly correlated spanner are similar to the results of the weakly correlated problem type. The rest of the problem types have similar results to that of subset sum. The results of subset sum is again similar to the results of NPP. In the *H1* landscape of the 0-1QKP like in the uncorrelated 0-1KP, the largest basins quickly covers large part of the

search space. In instances from the large $CV$ interval $[1, 2)$, large portion of the search space gets covered by very few basins, where only 10% of the basins cover between 70% to 90% of the search space. The same percentage of the basins cover around 50% to 70% of the search space in instances with $CV \in (0.3, 1)$, and cover around 50% of the search space in instances with $CV \in (0, 0.3]$. In all the problems, the search space is covered by fewer basins in the *H1* landscape compared to the *H1+2* one (in NPP, in instances with $CV \leq 0.3$, in particular). For instance, around 90% of the search space is covered by half of the basins in the *H1* landscape, while it takes around 70% of the basins to cover the same amount of the search space in the *H1+2* landscape.

### 6.2 Basin size and fitness

Another important aspect of the fitness landscape is the correlation between the basin size and the fitness of the optimum. Previous studies have shown that in general, fitter optima have larger basins (Stadler, 2002; Tayarani-N. and Prügel-Bennett, 2014), and landscapes with this kind of feature usually tend to be easier to search. Figure 7 shows the correlation between basin size and fitness of selected results of the three problems. We measured the correlation between the two quantities using Spearman's correlation coefficient instead of the traditional Pearson's correlation coefficient. The reason for that is Pearson's method assumes that both variables are drawn from normal distribution while Spearman's method is a non-parametric method. As we have seen in the previous section, the distribution of the fitness values and the basin sizes are highly skewed and far from being normally distributed. In NPP, the correlation between the objective and the basin size in both landscapes was found in general to be moderately negative $(0.4 - 0.6)$ to strongly negative $(> 0.6)$ (remember we are minimising here). If we excluded the *H1* with $CV \leq 0.3$, the correlation appears to remain more or less the same across the values of $k$ and $CV$ apart from few cases in the *H1+2* landscape. The correlation also appears to get slightly stronger with larger $n$. For the *H1* landscape of instances with $CV \leq 0.3$, the correlation seems to be slightly stronger when $k < k_c$ and it seems to get weaker with larger $n$ in the hard phase. In the 0-1KP, the correlation between the basin size and fitness was found to vary between weak to strong positive in both landscapes and seems not to change much across the values of $k$ or $CV$. The few exceptions and the most surprising ones are of instances of type inverse strongly correlated with $CV > 0.3$ and few other cases where the correlation was found to be negative. This negative correlation is in fact unusual in the combinatorial optimisation problems studied in the literature, where in general fitter optima were found have larger basins (ibid.). Similar to the uncorrelated 0-1KP problem types, the correlation between the attraction basin size and the optimum fitness in both landscape of the 0-1QKP is very strong and positive, indicating that indeed in this problem the fitter optima tend to have larger basins. This does not seem to change much across the $CV$ values for all the values of $\Delta$. However, as $\Delta$ gets larger the correlation in the *H1+2* landscapes of some of the instances seems to get weaker and sometimes even very strong negative (this is more noticeable in the small $CV$ interval $(0, 0.3]$). This indicates that, in such instances, fitter optima have smaller basins. Usually this feature means that these landscapes are more difficult to search particularly for local search, as the fitter optima has less probability of being found with a hill climber.

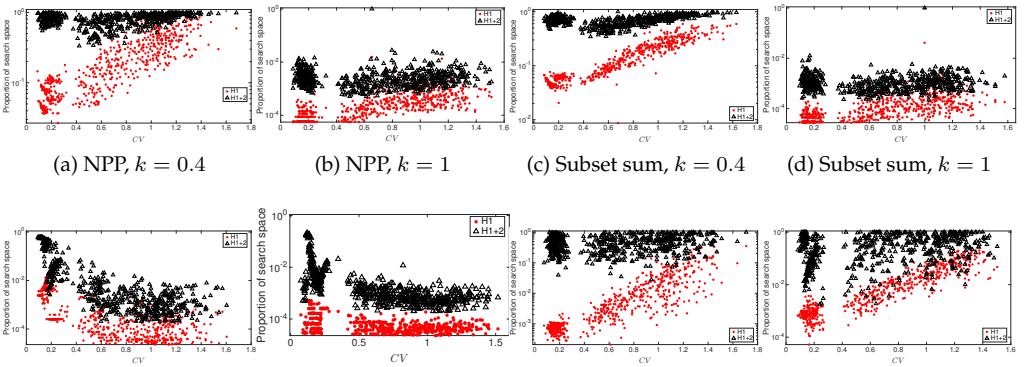


(a) NPP, $k = 0.4$ (b) NPP, $k = 1.2$ (c) Subset sum, $k = 0.4$ (d) Subset sum, $k = 1$

(e) Inverse strongly correlated, $k = 0.4$ (f) Inverse strongly correlated, $k = 1$ (g) 0-1QKP, $\Delta$ =0.1 (h) 0-1QKP, $\Delta$ =0.95

Figure 7: Spearman's rank correlation coefficient between basin size and fitness versus $CV$. The results are for 600 instances of size $n = 20$. $k = 1$ for the 0-1QKP.

### 6.3 Global Basin

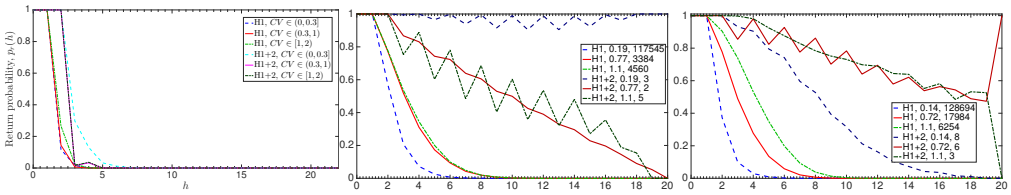


(a) NPP, $k = 0.4$ (b) NPP, $k = 1$ (c) Subset sum, $k = 0.4$ (d) Subset sum, $k = 1$

(e) Inverse strongly correlated, $k = 0.4$ (f) Inverse strongly correlated, $k = 1$ (g) 0-1QKP $\Delta$ =0.1 (h) 0-1QKP $\Delta$ =0.95

Figure 8: The basin size proportion (in log scale) of all global optima in an instance for each landscape against the $CV$. The results in each plot are for 600 instances of $n = 20$.

Figure 8 shows the global basins proportion out of the search space against the $CV$ for both landscapes and for some interesting instances of all the three problems. Clearly, this proportion translates to the probability of finding the optimal solution. In general and across all problems, the probability of finding the optimal solution is always higher in the *H1+2* landscape than in the *H1* one. In NPP and 0-1KP subset sum, the probability of finding the global optima in the *H1+2* landscape drops down from almost around $\sim 1$ in $k < k_c$ to around $\sim 10^{-2}$ in $k > k_c$. In *H1* landscape the probability of finding the global decreases from between around $\sim 0.1$ in small $CV$ instance and $\sim 0.8$ in large $CV$ instances when $k < k_c$, to reach around $\sim 10^{-4}$ and $\sim 15 \times 10^{-3}$ when $k > k_c$. The probability of finding the global when $k > k_c$, in both landscapes, decreases as the problem size grows. This is not surprising since the number of local optima in both problems grows exponentially with $n$. Like the previously studied features, the prob-




(a) NPP (b) 0-1KP Uncorrelated (c) 0-1QKP $\Delta$ =0.25

Figure 9: Return probability $p_r(h)$ to the global optimum starting from a Hamming sphere of radius $h$ (y-axis) versus $h$ (x-axis). The results are for 3 instances with $k = 1$ and of size $n = 20$ and for 0-1KP and 0-1QKP and $n = 22$ for NPP. Each legend entry in 0-1KP and 0-1QKP plots shows respectively: the landscape type, the instance $CV$ value, and the number of optima in that landscape of that instance.

ability does not seem to change much across the $CV$ values in the *H1+2* landscape, unlike the *H1* landscape, where the probability of finding the global increases with the $CV$ in both phases. This increase can be attributed to the decrease in the number of local optima in this landscape as the $CV$ increases, and the strong correlation between the basin size and fitness. In the inverse strongly correlated instances the opposite happens as the probability decreases when the $CV$ increases, despite the fact that the number of local optima decreases as the $CV$ increases. This reflects the results of the correlation between the basin size and the fitness in this problem type, where the correlation was found to be moderately to strongly positive in the small $CV$ interval but it starts to decease as $CV$ increases to a strong negative sometimes. We continue to see the effect of this feature on the performance of local search to find the global in this problem type in the next section. The probability of finding the global is very high in the *H1+2* landscape of the 0-1QKP and the uncorrelated 0-1KP. In the 0-1QKP, the probability of finding the global in the *H1+2* landscape seems to decrease slightly as $\Delta$ increases. This again is a reflection of the correlation between basin size and fitness that we have seen previously. The correlation was found to be strongly negative in some of these instances, which explains the decrease in the global basin size.

In an attempt to study the shape of the global basin, we plot in Figure 9, the proportion of the configurations that are part of its basin in every Hamming sphere of radius $h$ around it. In NPP, the plot shows the result for one of the two global found, as the same result applies to the other global due to the search space symmetry. The results are shown for three instances of size $n = 22$. Note that the global basin is not the largest in all of these instances. For example, in the instance with the smallest $CV$ value, the global basin proportion in the *H1* landscape is $1.98 \times 10^{-05}$ while the largest basin proportion is $1.86 \times 10^{-04}$. Similarly in the *H1+2* landscape the global basin proportion is $0.001$ while the largest basin proportion is $0.009$. We can see that in both landscapes the configurations in the global basin are concentrated in the immediate Hamming spheres around it. This is true for the both landscapes of all the problems apart from the *H1+2* landscape of the 0-1QKP, and 0-1KP the uncorrelated, weakly correlated, and uncorrelated spanner types, where the probability of returning to the global continues until the last sphere sometimes. Interestingly, we see that the return probability in some of the *H1+2* landscapes does not decreases monotonically but have an oscillating behaviour as $h$ increases. This can be attributed to the very small number of optima in such landscapes and the nature of the *H1+2* neighbourhood, as the neighbours of a configuration

in a given sphere $h$ would be spread over five spheres using this neighbourhood operator compared to only two spheres when using the *H1* operator. Figure 10 presents an example illustrating how this oscillating behaviour in the *H1+2* landscape can occur. In the 0-1QKP, we can see that the return to the global in the *H1+2* landscape of instances with small $CV$ values approaches zero faster in larger values of $\Delta$, where the return probability is almost zero in configurations different than the global in half or more of the dimensions. This agrees with the results obtained previously about the proportion of the global basins and the correlation between the fitness and the basin size.

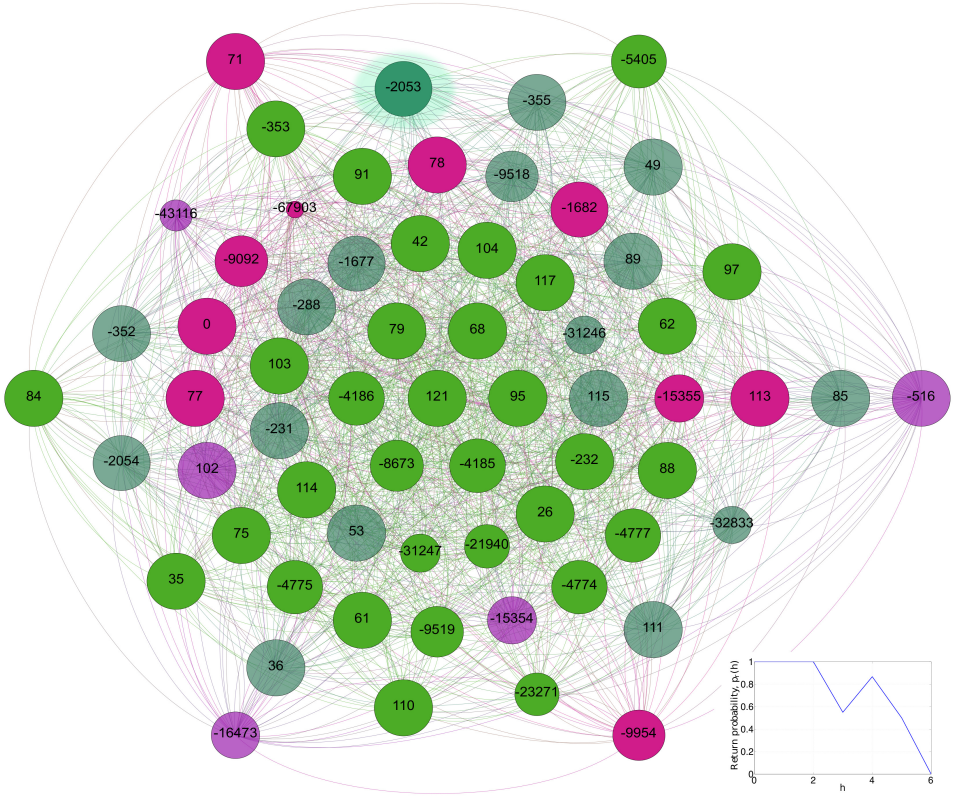
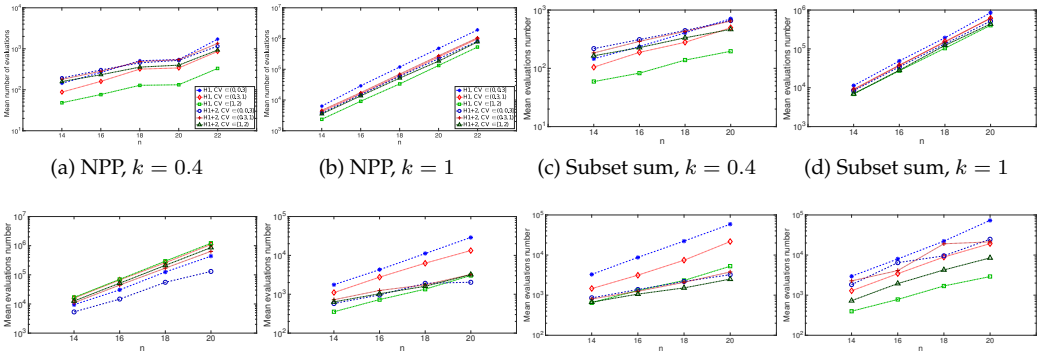



Figure 10: An example of an *H1+2* landscape where the return probability to the global starting from a Hamming sphere of radius $h$ does not decrease monotonically as $h$ increases. Each node in the graph represents a configuration and each edge indicates a neighbourhood relation. The objective value is shown for each node and is used to proportionally scale the node size. The graph has been laid out such that the global optimum (with objective value 121) is placed in the centre and the configurations that are $h$ Hamming distance away from the global lie on the $h$-th circle. The colours dark green and light green indicate that a configuration is in the global's basin, while the colours pink and purple indicate that a configuration is not in the global's basin. The semi-transparent nodes with the colours dark green and purple are neighbours of the highlighted node, with fitness -2053, in the 4th circle. In addition to the global, there are two strict local optima with fitness 102 and 113. The results are for an instance of weakly correlated knapsack of size $n = 6$, $k = 1$, $CV = 0.25$, and $\lambda = 0.5$.

## 7 Local Search Performance

### 7.1 Cost of finding the global and quality of optima obtained with fixed budget



(a) NPP, $k = 0.4$ (b) NPP, $k = 1$ (c) Subset sum, $k = 0.4$ (d) Subset sum, $k = 1$

(e) Inverse strongely correlated, $k = 1$ (f) Uncorrelated, $k = 1$ (g) 0-1QKP, $\Delta$ =0.1 (h) 0-1QKP, $\Delta$ =0.95

Figure 11: Number of fitness evaluations used to find the global (in log scale) against $n$. Each data point is an average of 30 runs of steepest ascent, averaged over the number of instances in each $CV$ interval. The results for each $n$ are for 600 instances.

We ran local search with random restarts until the optimal solution is found for 30 times per instance for $n \leq 22$. The cost of finding the global optima is then calculated using the number of used fitness evaluations. Note that we treat the objective function as a black-box here, hence the number of times the objective function is queried for each step taken equals the neighbourhood size of the employed operator. In NPP and the 0-1KP subset sum, the average number of fitness evaluations used to find the global increases as we approach the phase transition point and keep increasing as we cross it for all the different distributions and for both landscapes. This is expected due to the drastic decrease in the number of global optima in the phase with $k > k_c$. As we have seen before, the probability of finding the global is higher in instances with $k < k_c$, the algorithm quickly finds one of the many global optima while it struggles to find the single (two in NPP due to symmetry) global optimum when $k > k_c$. This indeed makes the phase transition in both of these problem a transition between "easy" to "hard" to solve for local search algorithms. In both the problems, the cost of finding the global optimum grows exponentially with $n$ in the hard phase as Figure 11 depicts. The growth in the easy phase seems to be much slower but we are unable to comment on its growth type as the trend is not very clear from the data. The cost of finding the global in the rest of the problems is the same across all values of $k$. In almost all the problems apart form the 0-1KP inverse strongly correlated and the 0-1QKP with $\Delta < 0.5$, the *H1* operator has the lowest mean number of used fitness evaluations to find the global in instances with $CV \in [1, 2)$. The uncorrelated and uncorrelated spanner instances have the lowest cost of finding the global in both landscapes. This is a reflection of the very strong positive correlation between the basin size and fitness, and the higher probability of returning to the global in theses instances. Instances of type inverse strongly correlated have the highest mean cost of finding the global, apart from the cost of the $CV \in (0, 0.3]$ interval. The increase in the cost in the instances with $CV > 0.3$ can be attributed to the strong negative correlation between the basin size and fitness, which

resulted in a lower probability of finding the global in these instances. In fact, we can see that the cost of finding the global using the *H1* operator in this problem type is the lowest in the $CV$ interval $(0, 0.3]$. Despite the fact that the number of local optima is the highest in this interval, which translates in all the other problem types to having the highest cost of locating the global out of all the $CV$ intervals. This again emphasises the importance of the correlation between the basin size and fitness (remember the correlation in the inverse strongly correlated was found to be moderately to strongly positive in $CV \in (0, 0.3]$ but it starts to decease as the $CV$ increases to a strong negative sometimes). The straight lines in almost all the observations in each problem type of the 0-1KP and the 0-1QKP indicate that the cost of finding the global seems to grow exponentially with $n$. In 0-1QKP, the cost of the *H1+2* operator increases slightly as $\Delta$ increases. In instances with $CV \in (0, 0.3]$, the increase is almost one order of magnitude. This can be attributed to the negative correlation between basin size and fitness and the decrease in the global basin proportion as $\Delta$ increases in these instances. The cost using the *H1+2* operator, in particular in instances with $CV \in (0, 0.3]$, is lower than that when *H1* is used. This might be surprising considering the quadratic (in $n$) cost associated with every *H1+2* step compared to the linear one in the *H1* case. This, however, can be explained by the very big difference in the number of local optima between the two landscapes, which suggest that the algorithm had to do far fewer number of restarts when using the *H1+2* operator. For the rest of the $CV$ intervals, the *H1* operator seems to have a lower cost even though it still has more local optima than that in the *H1+2* landscape. However, the difference between the two here is less by one order of magnitude, which makes it enough for the *H1+2* quadratic cost to offset the advantage of having lower number of local optima.

To examine further the relation between the number of fitness evaluations needed to explore the neighbourhood and the difference in the number of local optima between the two landscapes we compare the performance of the two operators in figure 12, where we determine the statistical significance between the two performances using Wilcoxon rank-sum test at the 5% level. In the NPP and subset sum, we can see that in the hard phase, the *H1* operator performs better when the $CV$ is large $\geq 1$. In the easy phase, the *H1* operator performs better across all the $CV$ intervals, despite the fact that the *H1+2* landscape is always smoother and has a higher probability of finding the global across all the values of the $CV$. This can be explained by the presence of many global optima in the easy phase which mitigates the ruggedness of the *H1* landscape, even in the very rugged landscape of the $(0, 0.3]$ $CV$ interval. However, this behaviour seems to fade away as $n$ grows in the sizes we studied in $n = 14 - 22$, where the *H1+2* operator starts to win more instances. This can be attributed to the growth of the number of local optima in the *H1* landscape in the $(0, 0.3]$ interval, which is the fastest growth rate out of all the $CV$ intervals, while the same interval has the lowest growth rate in the *H1+2* landscape. The trend of the *H1+2* operator performing better in the $(0, 0.3]$ $CV$ interval and the *H1* operator performing better in the $[1, 2)$ $CV$ interval continues to show in larger problem sizes of $n = 30, 100$ as shown in figure 12. Now the results show the quality of optima obtained by a fixed arbitrarily selected budget of fitness evaluations. As we have seen before in the NPP and subset sum, the difference in the quality between the two landscape decreases as the $CV$ grows, which explains the results for the wining case of the *H1* operator, of course alongside the lower difference in the number of local optima in this interval between the two landscape. Note that these results are specific to the budget we selected, whether the same trends will continue to occur with other budget values remains an open question. In the 0-1KP, apart

from subset sum, multiple strongly correlated, and profit ceiling types, the trends of the winner operator were found to change between when searching for the global and the quality of optima obtained by fixed budget search. They also were found to change with $n$. In some of the problem types, the number of tie cases seems to decrease as $n$ increases, and a clear winner emerges. In terms of the quality of the obtained optima with fixed budget search, the *H1* operator performs better in the strongly correlated, inverse strongly correlated and circle problem types, across all the $CV$ intervals. The *H1+2* operator performs better in the 0-1QK and the 0-1KP uncorrelated, weakly correlated, and their spanner equivalent across all $CV$ values. Theses results can be explained by the large difference in the quality of the optima between the two landscapes that we have seen in section 5.3, and the small number of optima and the faster decay of their proportions in the *H1+2* landscape.

### 7.2 Time to local optima

The time it takes steepest descent (ascent) starting from a random configuration until a local optimum is reached can reveal interesting aspects of the underlying structure of the landscape and particularly the shape of attraction basins. For each problem, we ran 1000 steepest descents per instance and collected the number of steps over 600 instances for each $n = 14, 16, 18, 20, 22$ and 500 instances for $n = 30, 100$. In the NPP and most problem types of the 0-1KP, this was found to be very small in both landscapes and grows very slowly with $n$ across all the $CV$ values. The medians of steps were between 1-3 when $n = 14$ (with the step size increasing with the $CV$) and only around 3 in $n = 100$ with outliers reaching up to 10 in $n = 14$ and up to 20 in $n = 100$. This can be partially attributed to the fact that the attraction basin sizes in both landscapes were found to be mainly small. In the *H1* landscape, we believe that this is additionally due to the large number of local optima in this landscape. In the *H1+2* landscape, even though the number of optima is much smaller than that in the *H1*, its neighbourhood is larger and its nature can allow it to take fewer number of steps by hopping over spheres to reach the local optimum, which explains the small number of steps in this landscape. The very slow growth of steps with $n$ can be attributed to the exponential growth of the number of local optima in both landscapes and across all the $CV$ values. The steps in the *H1* landscape of the 0-1QKP, and the 0-1KP uncorrelated and the uncorrelated spanner instances was found to be similar to the previous described results but slightly higher, outliers around 30 in $n = 100$, which is believed to be due to the larger basin sizes in these instances. However, the number of steps in the *H1+2* landscape of these problems and the weakly correlated, weakly correlated spanner, strongly correlated spanner, and multiple strongly correlated problem types, seems to grow faster with $n$, evident in the median steps being between 10 and 25 when $n = 100$. This supports our observation that the decay of the number of local optima in the *H1+2* landscape of these problem types is faster than that in the rest of the problems. Note that although the local search algorithm under study here considers only the best improving move, that does not necessarily guarantee that the path, starting from a random configuration $x$ until a local optimum $x^*$ is found, will be the shortest path (i.e. the number of steps taken from $x$ until $x^*$ is reached is at most equal to the Hamming distance between $x$ and $x^*$). From the perspective of the Hamming spheres around an optimum $x^*$, this can occur when the best improving move of a configuration $x^i$ in the path, which resides in the Hamming sphere $h^i$, is in a Hamming sphere $h^{i+1} \geq h^i$. The number of steps taken in the *H1* landscape was found to be always equal to the Hamming distance between the initial random configuration and the found local optimum. In the *H1+2* landscape

this was found to be almost always smaller or equal to the Hamming distance with the exception of extremely few cases where it was found to be one or five steps larger than the Hamming distance (in $n = 22$).

## 8 Conclusions

This paper presents an extensive landscape analysis of three NP-hard binary packing problems: the number partitioning problem (NPP), the binary knapsack problem (0-1KP), and the quadratic binary knapsack problem (0-1QKP). We performed a comparative study of the landscape induced by two neighbourhood operators, the *H1* operator with a neighbourhood that grows linearly with the problem size and a larger neighbourhood operator *H1+2* that has a quadratic growth neighbourhood. The set of properties that we studied includes: types of search position, number of local and global optima and plateaux, quality of optima and plateaux, basin size and its correlation with fitness, time to local optima, cost of finding the global solution, and quality of optima obtained with a fixed budget search. Our work focuses on studying how these properties vary with different values of problem parameters. Most of the existing studies of these problems only consider instances where the weights are drawn at random from a uniform distribution. We studied instances generated by drawing the weights at random from various distributions. In all of the three problems, we found that the number of strict local optima and the cost of the local search to find the global, vary greatly (most noticeably in the *H1* landscape) between some of the distributions. We proposed and demonstrated that the use of the $CV$ of the weights, a single parameter that is easy to calculate and does not require the knowledge of the underlying distribution of the weights, captures most of this variability. In all the investigated problems, there is a very strong and negative correlation between the $CV$ and the number of local optima in the *H1* landscape. We continued to see this trend in all the problem sizes we studied. The average number of local optima in this landscape seemed to be well approximated by the formula $\frac{a}{e^{bCV}}2^n$, where $a$ and $b$ depend on the problem and $n$. We believe that this phenomenon is particular to the binary packing problems related to the 0-1 knapsack problem. Interestingly, we did not find any significant difference between landscapes of strong (0-1QKP) and weak (0-1KP) NP-hard problems. This raises an important question concerning the difference between strong and weak NP-hard problems from the point of view of meta-heuristics.

Only the NPP and the subset sum problems, which is a generalisation of the NPP and a special case of the 0-1KP, have an identified phase transition. The only two properties that were found to change when the problem crosses the phase transition in these two problems are the number of global optima (and consequently the probability of finding the global) and the number of plateaux. The rest of the properties remained oblivious to the phase transition. This result is in agreement with the results obtained by Stadler et al. (2003), in which they found that the features of the uniform NPP landscape, that have been mapped into barriers trees, are insensitive to the phase transition. However, recently Ochoa et al. (2017) noted that the global funnel structure of the uniform NPP landscape does change with the phase transition. The performance of local search algorithms was found to be affected by the phase transition in both NPP and subset sum, where a considerable increase in the cost of locating the global solution occurs in instances with $k > k_c$, confirming that it is indeed a transition between "easy-to-solve" to "hard-to-solve" for local search algorithms.

## Acknowledgement

This work was supported by King Saud University, Riyadh, Saudi Arabia and partially supported by the Engineering and Physical Sciences Research Council [grant number EP/N017846/1].

## References

Alyahya, K. and Rowe, J. (2014). Phase transition and landscape properties of the number partitioning. In *14th European Conference on Evolutionary Computation in Combinatorial Optimisation, EvoCOP*, volume 8600 of *Lecture Notes in Computer Science*, pages 206–217.

Alyahya, K. and Rowe, J. (2016). Simple random sampling estimation of the number of local optima. In *Parallel Problem Solving from Nature - PPSN XIV*, volume 9921 of *Lecture Notes in Computer Science*, pages 932–941.

Borgs, C., Chayes, J., and Pittel, B. (2001). Phase transition and finite-size scaling for the integer partitioning problem. *Random Structures & Algorithms*, 19(3-4):247–288.

Burke, E., Curtois, T., Kendall, G., Hyde, M., Ochoa, G., and Vazquez-Rodriguez, J. (2009). Towards the decathlon challenge of search heuristics. In *proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers*, pages 2205–2208.

Caccetta, L. and Kulanoot, A. (2001). Computational aspects of hard knapsack problems. *Nonlinear Analysis: Theory, Methods & Applications*, 47(8):5547–5558.

Consoli, P. A., Minku, L. L., and Yao, X. (2014). Dynamic selection of evolutionary algorithm operators based on online learning and fitness landscape metrics. In *Simulated Evolution and Learning: 10th International Conference, SEAL 2014, Dunedin, New Zealand, December 15-18, 2014. Proceedings*, pages 359–370, Cham. Springer International Publishing.

Daolio, F., Liefooghe, A., Verel, S., Aguirre, H., and Tanaka, K. (2015). Global vs local search on multi-objective NK-landscapes: Contrasting the impact of problem features. In *proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*, pages 369–376.

Ferreira, F. and Fontanari, J. (1998). Probabilistic analysis of the number partitioning problem. *Journal of Physics A: Mathematical and General*, 31(15):3417.

Gallo, G., Hammer, P., and Simeone, B. (1980). Quadratic knapsack problems. In *Combinatorial Optimization*, volume 12 of *Mathematical Programming Studies*, pages 132–149.

Garey, M. and Johnson, D. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Series of books in the mathematical sciences. W. H. Freeman.

Gottlieb, J. (2001). On the feasibility problem of penalty-based evolutionary algorithms for knapsack problems. In *Applications of Evolutionary Computing*, volume 2037 of *Lecture Notes in Computer Science*, pages 50–59.

Grover, L. (1992). Local search and the local structure of NP-complete problems. *Operations Research Letters*, 12(4):235–243.

He, J., Reeves, C., Witt, C., and Yao, X. (2007). A note on problem difficulty measures in black-box optimization: Classification, realizations and predictability. *Evolutionary Computation*, 15(4):435–443.

Hooker, J. (1995). Testing heuristics: We have it all wrong. *Journal of Heuristics*, 1(1):33–42.

Hoos, H. and Stützle, T. (2005). *Stochastic Local Search: Foundations & Applications*. The Morgan Kaufmann Series in Artificial Intelligence. Elsevier Science.

Khuri, S., Bäck, T., and Heitkötter, J. (1994). The zero/one multiple knapsack problem and genetic algorithms. In *proceedings of the 1994 ACM Symposium on Applied Computing*, pages 188–193.

Martello, S., Pisinger, D., and Toth, P. (1999). Dynamic programming and strong bounds for the 0-1 knapsack problem. *Manage. Sci.*, 45(3):414–424.

Mertens, S. (2001). A physicist's approach to number partitioning. *Theoretical Computer Science*, 265(1-2):79–108.

Michalewicz, Z. and Arabas, J. (1994). Genetic algorithms for the 0/1 knapsack problem. In *Methodologies for Intelligent Systems*, volume 869 of *Lecture Notes in Computer Science*, pages 134–143.

Ochoa, G., Veerapen, N., Daolio, F., and Tomassini, M. (2017). Understanding phase transitions with local optima networks: Number partitioning as a case study. In *Evolutionary Computation in Combinatorial Optimization: 17th European Conference, EvoCOP 2017, Amsterdam, The Netherlands, April 19-21, 2017, Proceedings*, pages 233–248, Cham. Springer International Publishing.

Olsen, A. (1994). Penalty functions and the knapsack problem. In *proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*, pages 554–558.

Pisinger, D. (1999). Core problems in knapsack algorithms. *Operations Research*, 47(4):570–575.

Pisinger, D. (2005). Where are the hard knapsack problems? *Comput. Oper. Res.*, 32(9):2271–2284.

Pisinger, D. (2007). The quadratic knapsack problem–a survey. *Discrete Applied Mathematics*, 155(5):623–648.

Prügel-Bennett, A. and Tayarani-N., M.-H. (2012). Maximum satisfiability: Anatomy of the fitness landscape for a hard combinatorial optimization problem. *IEEE Transactions on Evolutionary Computation*, 16(3):319–338.

Reeves, C. and Eremeev, A. (2004). Statistical analysis of local search landscapes. *Journal of the Operational Research Society*, 55(7):687–693.

Sasamoto, T. (2003). Phase transitions of subset sum and shannon's limit in source coding. *Physica A: Statistical Mechanics and its Applications*, 321(12):369–374.

Sasamoto, T., Toyoizumi, T., and Nishimori, H. (2001). Statistical mechanics of an NP-complete problem: subset sum. *Journal of Physics A: Mathematical and General*, 34(44):9555.

Slaney, J. and Walsh, T. (2001). Backbones in optimization and approximation. In *International Joint Conference on Artificial Intelligence*, pages 254–259.

Smith-Miles, K. and Lopes, L. (2012). Measuring instance difficulty for combinatorial optimization problems. *Computers  Operations Research*, 39(5):875–889.

Stadler, P. (2002). Fitness landscapes. In *Biological Evolution and Statistical Physics*, volume 585 of *Lecture Notes in Physics*, pages 183–204.

Stadler, P., Hordijk, W., and Fontanari, J. (2003). Phase transition and landscape statistics of the number partitioning problem. *Physical Review E*, 67(5):056701.

Stadler, P. and Stephens, C. (2002). Landscapes and effective fitness. *Comments on Theoretical Biology*, 8(4-5):389–431.

Swan, J., Adriaensen, S., Bishr, M., Burke, E., Clark, J., Causmaecker, P. D., Durillo, J., Hammond, K., Hart, E., Johnson, C., Kocsis, Z., Kovitz, B., Krawiec, K., Martin, S., Merelo, J., Minku, L., Ozcan, E., Pappa, G., Pesch, E., Garcia-Sanchez, P., Schaerf, A., Sim, K., Smith, J., Stutzle, T., VoB, S., Wagner, S., and Yao, X. (2015). A research agenda for metaheuristic standardization. In *MIC 2015: The 11th Metaheuristics International Conference*.

Srensen, K. (2015). Metaheuristicsthe metaphor exposed. *International Transactions in Operational Research*, 22(1):3–18.

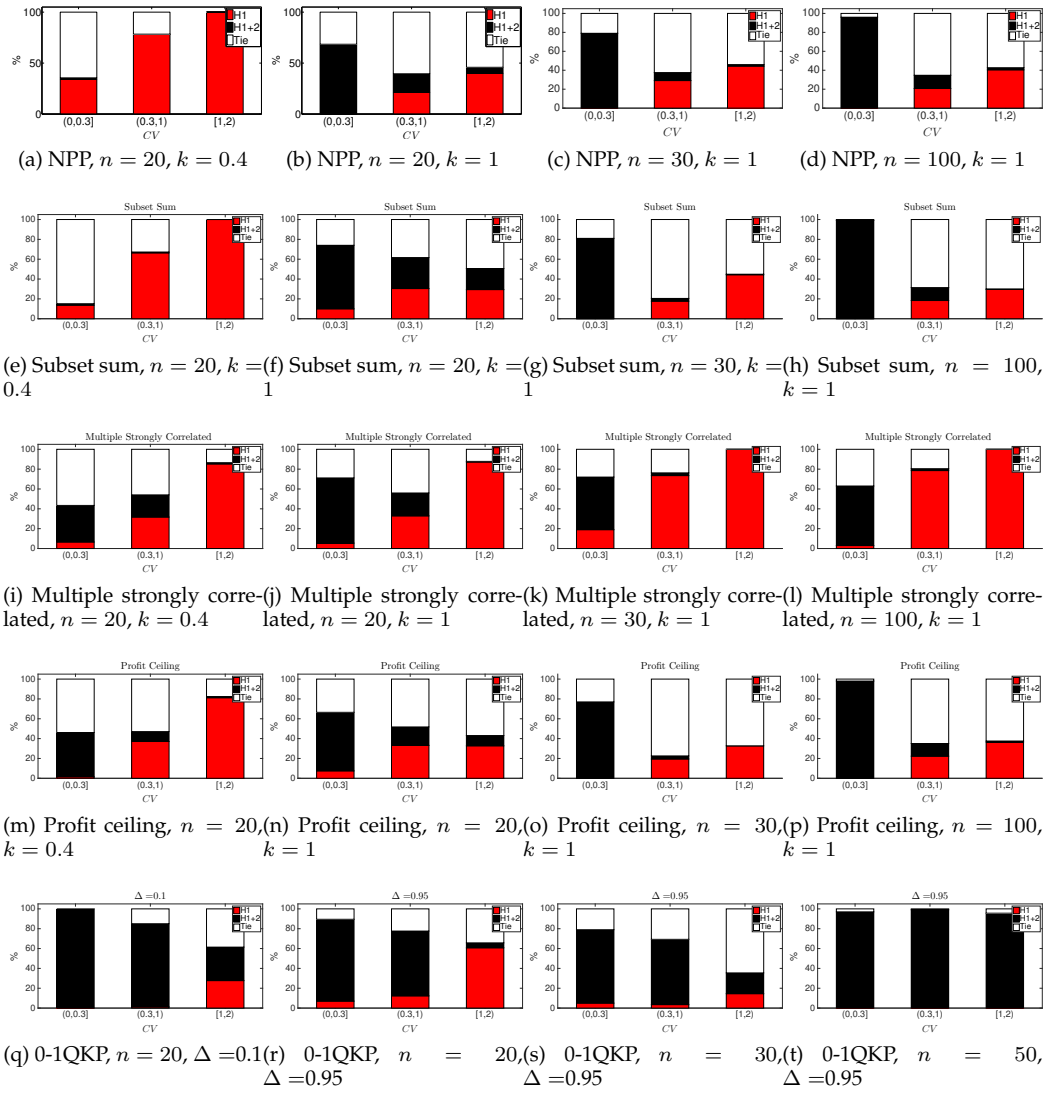
Tayarani-N., M.-H. and Prügel-Bennett, A. (2014). On the landscape of combinatorial optimization problems. *IEEE Transactions on Evolutionary Computation*, 18(3):420–434.

Tayarani-N., M.-H. and Prügel-Bennett, A. (2015a). An analysis of the fitness landscape of travelling salesman problem. *Evolutionary Computation*, pages 1–38.

Tayarani-N., M.-H. and Prügel-Bennett, A. (2015b). Anatomy of the fitness landscape for dense graph-colouring problem. *Swarm and Evolutionary Computation*, 22:47–65.

Tayarani-N., M.-H. and Prügel-Bennett, A. (2015c). Quadratic assignment problem: a landscape analysis. *Evolutionary Intelligence*, 8(4):165–184.

Whitley, D., Sutton, A., and Howe, A. (2008). Understanding elementary landscapes. In *proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*, pages 585–592.




(a) NPP, $n = 20$, $k = 0.4$ (b) NPP, $n = 20$, $k = 1$ (c) NPP, $n = 30$, $k = 1$ (d) NPP, $n = 100$, $k = 1$

(e) Subset sum, $n = 20$, $k = 0.4$ (f) Subset sum, $n = 20$, $k = 1$ (g) Subset sum, $n = 30$, $k = 1$ (h) Subset sum, $n = 100$, $k = 1$

(i) Multiple strongly correlated, $n = 20$, $k = 0.4$ (j) Multiple strongly correlated, $n = 20$, $k = 1$ (k) Multiple strongly correlated, $n = 30$, $k = 1$ (l) Multiple strongly correlated, $n = 100$, $k = 1$

(m) Profit ceiling, $n = 20$, $k = 0.4$ (n) Profit ceiling, $n = 20$, $k = 1$ (o) Profit ceiling, $n = 30$, $k = 1$ (p) Profit ceiling, $n = 100$, $k = 1$

(q) 0-1QKP, $n = 20$, $\Delta = 0.1$ (r) 0-1QKP, $n = 20$, $\Delta = 0.95$ (s) 0-1QKP, $n = 30$, $\Delta = 0.95$ (t) 0-1QKP, $n = 50$, $\Delta = 0.95$

Figure 12: The percentage of instances where each operators performed significantly better and the percentage where no significance difference was found (Tie). Significance determined using Wilcoxon rank-sum ($p-$value $\leq 0.05$). The results are for 600 instances for $n = 20$ and 500 for $n = 30, 50, 100$. Starting from the left, the first two columns show the number of fitness evaluations used to find the global optimum averaged over 30 runs per instance. The remaining two columns show the quality of optima found with a fixed budget search. The quality of the solution found is averaged over 1000 runs of local search with fixed budget of $10^5$ fitness evaluations.