Landscape Analysis of a Class of NP-Hard Binary Packing Problems
- SUPPLEMENTARY MATERIAL -

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

Jonathan E. Rowe
School of Computer Science, University of Birmingham, B15 2TT, UK
J.E.Rowe@cs.bham.ac.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
1 Number Partitioning Problem

Figure 1: Violin plots showing the number of the different optima and plateaux found in the $H1$ landscape against $k$. The results are for 600 instances of $n = 20$ for each value of $k$. The colours show the different ranges of $CV$ values, A violin plot is a mixture of a box plot and a kernel density plot. In addition to the usual four main features shown by a box plot (i.e. centre, spread, asymmetry and outliers), violin plot adds an estimated density trace (smoothed histogram), which reveals the shape of the data distribution that would not have been obvious in a box plot otherwise (Hintze and Nelson, 1998).
Figure 2: Sizes of the different plateaux and the number of exits in open plateaux found in the $H1$ landscape. The results are for 600 instances of $n = 20$ for each value of $k$. The colours show the different ranges of $CV$ values. All open plateaux found in the $H1$ landscape are composed of a single configuration.

Figure 3: Sizes of the different plateaux and the number of exits in open plateaux found in the $H1+2$ landscape. The results are for 600 instances of $n = 20$ for each value of $k$. The colours show the different ranges of $CV$ values.
Figure 4: Number of the different optima and plateaux found in the $H1+2$ landscape. The results are for 600 instances of $n = 20$ for each value of $k$. The colours show the different ranges of $CV$ values.
Figure 5: The different types of optima and plateaux found in an easy NPP instance \((k = 0.5)\) of size \(n = 12\) and \(CV = 0.61\). The colours of the nodes and their labels correspond to their type as follows: (1, red) a strict global optimum, (2, pink) a global plateau, (3, green) a strict local optimum; (4, blue) a closed plateau; (5, dark grey) an open plateau; and (6, light grey) an exit. The node size is scaled proportional to its fitness (larger means fitter). An edge between two nodes can either indicate that they are neighbours, or if it is between an exit and a node with a better fitness, it indicates that the exit leads to the basin of that optimum or plateau.
Figure 6: (Left) The mean and the standard deviation of the fitted log-normal distributions to the basin sizes against the $CV$. The results are for both landscape of 600 instances for each $n = 20, 22$ and $k = 1$. (Right) Histograms of the basin sizes of two instances with small and large $CV$ values; the fitted log-normal distribution is shown for each landscape.
Figure 7: The quality of optima and plateaus in the $H1$ and $H1+2$ landscapes. The x-axis shows the fitness value divided by $\sum_{i=1}^{n} w_i$. The data includes all optima and plateaux found in 600 instances for each $k$ value of problem size $n = 20$. 

$k = 0.4$

$k = 1$

$k = 1.2$
Figure 8: The quality of optima in the $H1$ and $H1+2$ landscapes ($k = 1$). The x-axis shows the fitness value divided by $\sum_{i=1}^{n} w_i$. The data includes all sampled optima from 500 instances for each $n$. The sampling for each instance includes 1000 steepest descents and SRS of size $s = 10^5$, $5 \times 10^5$ for $n = 30, 100$ respectively.
2 0-1 Knapsack Problem

2.1 Problem Types

Given a positive integer $a$ and $w_i$ drawn at random from a given data range $[1, M]$, the profit $p_i$ can be expressed as a function of $w_i$ yielding the following instance types:

**Uncorrelated** $\text{ucorr}$: there is no correlation between the profit and weight of an item; $p_i$ is uniformly random in $[1, M]$.

**Weakly Correlated** $\text{wcorr}$: despite the label of this instance type, the profit and weight of an item are highly correlated; $p_i$ is chosen uniformly at random from $[w_i - M/a, w_i + M/a]$ such that $p_i \geq 1$. 

Figure 9: The relation between weights (x-axis) and profits (y-axis) in the different types of knapsack instances. The different colours indicate different $CV$ values of the weights. The weights have been shifted along the x-axis for better visibility. (a) Uncorrelated. (b) Weakly correlated. (c) Strongly correlated. (d) Inverse strongly correlated. (e) Subset sum. (f) Uncorrelated spanner span$(2, 10)$. (g) Weakly correlated spanner span$(2, 10)$. (h) Strongly correlated spanner span$(2, 10)$. (i) Multiple strongly correlated mstr$(3M/10, 2M/10, 6)$. (j) Profit ceiling $pcei1(3)$. (k) Circle $circle(2/3)$.
**Strongly Correlated** \( \text{scorr} \): the profit of an item is linearly related to its weight \( p_i = w_i + M/a \).

**Inverse Strongly Correlated** \( \text{invs corr} \): like strongly correlated instances, the profit of an item is linearly related to its weight but with a negative fixed charge; \( p_i = w_i - M/a \), and \( w_i \) is drawn at random from \([M/a + 1, M(M/a)]\). In the original definition of this instance, the weights were assigned accordingly after the profits have been sampled. We changed the definition slightly by sampling the weights first, to preserve the \( CV \) value of the weights.

**Subset Sum** \( \text{sbst sum} \): the item’s profit and weight are equal \( p_i = w_i \). Obtaining a filled knapsack is thus the only aim when solving instances of this type.

The previous types are standard instances in the literature of the 0-1KP. The following instance types were proposed by Pisinger in Pisinger (2005). They are constructed in such a way to make them difficult for the branch-and-bound algorithms.

**Spanner** \( \text{span}(v, d) \): a set called the spanner set is generated with \( v \) items each with a profit and a weight. The spanner type is characterised by \( v \), the spanner set size, and \( d \), a multiplier limit. The weight of each item is drawn at random from a given range \([1, M]\). The profit of each item is then generated according to the distribution of the spanner problem type to be: *uncorrelated* (uspan), *weakly correlated* (wspan), or *strongly correlated* (span) with the item’s weight. The items in the spanner set are then normalised by dividing both profits and weights by \( d + 1 \). The last step is to construct the \( n \) items by randomly selecting an item \((w_i, p_i)\) from the spanner set and a multiplier \( b \) drawn from the interval \([1, d]\) such that the constructed item has the following profit and weight \((bw_i, bp_i)\). All items in a spanner instance are multiples of the spanner set. For all the spanner problem types, \( v \) and \( d \) are set to equal 2 and 10. Note that because of the way the spanner problem types are generated, it was difficult to generate instances of this type with \( CV \leq 0.3 \).

**Multiple Strongly Correlated** \( \text{mstr}(k_1, k_2, d) \): if the weight \( w_i \) is divisible by \( d \), then \( p_i := w_i + k_1 \), otherwise, \( p_i := w_i + k_2 \). Since the weights in the first group are all multiples of \( d \), using only these weights will fill at most \( d\lfloor C/d \rfloor \) of the capacity Pisinger (2005). Hence, the need to use some of the items from the second distribution to obtain a completely filled knapsack. In both groups the profits and weights are strongly correlated. Setting the values of \((k_1, k_2, d)\) to \((3M/10, 2M/10, 6)\) has been shown to generate very difficult instances according to computational experiments in Pisinger (2005).

**Profit Ceiling** \( \text{pceil}(d) \): all profits are multiples of a given parameter \( d \), \( p_i = d\lceil w_i/d \rceil \).

We set \( d \) to 3, since this setting produces difficult instances according to Pisinger (2005).

**Circle** \( \text{circle}(d) \): the item’s profit is a function of the weights from an arc of an ellipsis, \( p_i := d\sqrt{4M^2 - (w_i - 2M)^2} \) and \( d = 2/3 \).
2.2 Results

Figure 10: The basin size fractions of the search space using different penalty functions. The red colour corresponds to the feasible configurations that are part of the basin and the black corresponds to the infeasible ones. The figures show the global optimum basin and the largest 20 local optima basins. The optima are ordered according to fitness (x-axis) starting from the global optimum in the far left. The results are for an instance of weakly correlated knapsack of size $n = 22$, $k = 1$, $CV = 0.12$, and $\lambda = 0.5$. The fraction of the number of optima in the $H_1$ landscape is 0.115 and in the $H_1+2$ landscape is $1.5 \times 10^{-6}$. The correlation between fitness and basin size of all the optima for each landscape is shown by Kendall’s $\tau$, Pearson’s $r$, and Spearman’s $r_s$ correlation coefficients. Obviously changing the penalty function only affects the infeasible part of the basin (the black region of the bars).
K. Alyahya, J.E. Rowe

Uncorrelated

Weakly correlated

Strongly correlated

Inverse strongly correlated
Figure 11: The quality of optima and plateaus 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 x-axis shows the fitness value divided by $\sum_{i=1}^{n} p_i$. The data includes all optima and plateaux found in 600 instances for each problem type of problem size $n = 20$ and $k = 1$. 
3 Quadratic 0-1 Knapsack Problem

\[ \Delta = 0.1 \]

\[ \Delta = 0.25 \]

\[ \Delta = 0.5 \]

\[ \Delta = 0.75 \]
Figure 12: The quality of optima and plateaus 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 x-axis shows the fitness value divided by $\sum_{i=1}^{n} \sum_{j=1}^{n} p_{ij}$. The data includes all optima and plateaux found in 600 instances for each $\Delta$ of problem size $n = 20$ and $k = 1$.

References
