24th European Conference on Artificial Intelligence - ECAI 2020
Santiago de Compostela, Spain

Is hardness inherent in computational problems? Performance of human and electronic computers on random instances of the 0-1 knapsack problem

Nitin Yadav [1] and **Carsten Murawski**[2] and **Sebastian Sardina**[3] and **Peter Bossaerts** [4]

Abstract. Many cognitive problems people face have been shown to be computationally intractable. However, tractability is typically defined in terms of asymptotic worst-case behaviour of instances. One approach for studying typical cases of NP-complete problems is based on random instances. It has been shown that random instances of many NP-complete problems exhibit a phase transition in solvability and that hard instances tend to occur in this phase transition. Here, we characterise a phase transition in solvability for random instances of the 0-1 knapsack problem in terms of two simple instance properties. Subsequently, we show that compute time of algorithms peaks in the phase transition. Remarkably, the phase transition likewise predicts where people spend the most effort. Nevertheless, their performance decreases. This suggests that instances that are difficult for electronic computers are recognized as such by people, but the increased effort does not compensate for hardness. Given the ubiquity of the knapsack problem in every-day life, a better characterisation of the properties that make instances hard will help understand commonalities and differences in computation between human and digital computers, and to improve both decision environments (contracts, regulation) as well as human-computer interfaces.

1 Introduction

Previous research in artificial intelligence (AI) has shown that random instances of many NP-complete problems have a phase transition in solvability (e.g., [17, 7, 6]). We build on this work to *analyse human reasoning* and its *relation with computational complexity*. Concretely, we investigate a phase transition—a sudden change in solvability of random instances—in the well-known *0-1 Knapsack Problem* (KP) [15, 13] and how it is related to instance difficulty for human decision-makers. Combining theoretical analysis and experimental evaluation, we *(a)* identify and characterise a systematic transition from solvability to unsolvability using two easily observable instance properties; *(b)* demonstrate that, as has been shown for other NP-complete problems, the computationally hard instances happen to lie on the boundary of the phase transition; and *(c)* provide strong evidence that instances that are challenging for people occur in the same region. Our findings are significant in that understanding the computational complexity of instances is relevant for many aspects of human behavior [20], given that the KP is encountered at many different levels of cognition, including attention [26], intellectual discovery [16] and investment decisions [14].

In computer science (CS) and AI, significant work has been carried out to understand how difficult a problem is for computers to solve. Problem complexity provides characterizations of lower bounds of resource requirements (e.g., time or space): *any* algorithm will require at least those resources to solve the problem. Such characterizations are, however, often too coarse and conservative, in that they are based on the worst-case behaviour of algorithms, which may not coincide with ‘typical’ case instances. A more detailed analysis, which we adopt here, involves studying the relation between *normalised properties* and *normalised complexity* of a problem. In AI research, studying phase transitions in solvability of random instances is considered one approach to gain a better, more meaningful, understanding of instance complexity [12]. Indeed, such phase transitions in solvability have been documented for several NP-complete problems, such as SAT [17], Hamiltonian Circuit [7], and Integer Partitioning [6], as well as for many physical systems, with complexity peaking at the phase boundary [18]. The question we ask here is *whether we apply these insights to gain a better understanding of hardness of instances for human reasoning?*

Even though the nature of computation in the human brain (analogue, parallel and error-prone) likely is fundamentally different from that of an electronic computer [22], there may exist structural properties in problems that may affect both human and electronic computers in similar ways. Since human computation could involve techniques such as greedy approximation, randomisation, hit and trial, etc., it is worth investigating the relevance of phase transitions in solvability for human reasoning. Here we show that there may in fact exist structural properties of problems that affect people's ability to solve instances as much as computers'. An understanding of such properties (e.g., those that arise from computational complexity) will benefit research on human computation within artificial intelligence. With a growing interest in AI software with the *human in-and-on the loop*, understanding when a task may be difficult for a person becomes necessary for building effective human-centric intelligent systems.

Thus, given the ubiquity of the KP in every-day life, we study here the difficulty of the decision variant of the (0-1) KP for both electronic and human computers. To that end, we first identify a region where KP displays a systematic transition from solvability to unsolvability within a narrow range of two readily observable instance parameters: *instance* capacity and *instance* profit. We provide a theoretical analysis (Section 3) that utilizes a fast greedy-type estimation rule to provide both a lower and an upper bound of the probability

[1] The University of Melbourne, email: nitin.yadav@unimelb.edu.au
[2] The University of Melbourne, email: carstenm@unimelb.edu.au
[3] RMIT University, email: sebastian.sardina@rmit.edu.au
[4] The University of Melbourne, email: peter.bossaerts@unimelb.edu.au

that a given instance admits a solution, and confirm the phenomena via computational analysis (Section 4). In our analysis, we show an interesting "convexity" property of the phase transition region. Secondly, we demonstrate that, similar to other NP-complete problems, the computationally hard instances happen to lie on the boundary of the phase transition (Section 4). Thirdly, and possibly most interesting to the reader, we provide empirical evidence that suggests that instances that are challenging for people also happen to occur near the identified phase transition (Section 5).

Two important observations are worth noting. First, our findings complement our previous work [20] where we demonstrated that human performance in the KP (optimisation variant) is correlated with instance complexity. However, this work uses an algorithm-specific property, namely the "Sahni-k" [24] metric, to provide an *ex-post* complexity measure of KP instances. In contrast, here we identify an *ex-ante* complexity measure that has been empirically shown to be algorithm-independent. Second, we note that we are not the first to study instance difficulty of the KP (e.g., [23, 25, 13]). However, previous work has focused primarily on the optimisation variant of the KP and, in particular, on studying the performance of specific algorithms tailored for different optimisation cases. In particular, this work uses correlation between weights and values of items to characterise complexity of instances for specific algorithms, but these measures do not generalise. Importantly, these studies do not identify any phase transition or explore its relation with computational and human reasoning, which is the main objective of our work.

2 The Knapsack Problem

The 0-1 Knapsack Problem (KP) is a combinatorial optimisation problem with the goal of finding the subset of a set of items with given values and weights that maximises total profit subject to a capacity (weight) constraint. The KP is NP-hard. The number of subsets that can be formed from the n available items increases exponentially in n (2^n). However, as is the case for some NP-hard problems [17], many instances can be solved in polynomial time, even if n is large. It has been shown for some NP-complete problems that hardness appears to be closely related to phase transitions in solvability of instances [3, 19], which typically occurs within a narrow range of instance properties. It is an open question whether this is also the case for the KP.

To address this question, we consider here the *decision variant* of the KP: *Does there exists a subset of items from the set of available items with total profit at least as high as a given profit constraint, and total weight at most as high as a given capacity constraint?* Formally, given a set of items $I = \{1, \ldots, n\}$ with weights $\langle \hat{w}_1, \ldots, \hat{w}_n \rangle$ and values $\langle \hat{v}_1, \ldots, \hat{v}_n \rangle$, where each $\hat{w}_i$ and $\hat{v}_i$ is a positive integer, and two positive integers $\hat{c}$ and $\hat{p}$ denoting the capacity and profit constraint (of the knapsack), *decide whether there exists* a knapsack set $S \subseteq I$ such that:

- $\sum\limits_{i \in S} \hat{w}_i \leq \hat{c}$, the weight of the knapsack is less than or equal to the capacity constraint; and
- $\sum\limits_{i \in S} \hat{v}_i \geq \hat{p}$, the value of the knapsack is greater than or equal to the profit constraint.

The decision variant of the KP is closely related to the optimisation variant; the latter can be solved by iteratively solving the former while incrementing the profit constraint.

The KP is a constrained satisfaction problem with two opposing constraints. Increasing the profit level requires adding items to the knapsack, while decreasing the weight requires removing items from the knapsack.

We use a small example to highlight this tension between the two constraints. Consider an instance of the knapsack problem consisting of 4 items with weights $\langle 2, 5, 8, 4 \rangle$ and profits $\langle 3, 2, 6, 9 \rangle$. The panels in Figure 1 show the search space for this instance (for two different capacity and profit constraints). Each node in the graph is a knapsack configuration and the edges link the knapsack configurations that are reachable by addition or deletion of a single item. For example, the node $\{1, 3\} : 10 : 9$ represents a knapsack containing two items 1 and 3, with a total weight of 10 and total profit of 9. This node can be reached in 4 ways (there are 4 edges linking this node), one of which is to add item 3 to a knapsack that already contains item 1. The layout of the graph is such that, as we traverse from left to right, the number of items in the knapsack increases. The vertical alignment is representative of the number of subsets of fixed cardinality (i.e., $\binom{n}{k}$). We use the empty knapsack (left most node) as the initial node of the search space.

Suppose the capacity and the profit constraints for this instance are 10 and 15, respectively. This is the case in the example displayed in Figure 1a. Triangle nodes represent configurations that satisfy the capacity constraint, whereas square nodes denote configurations that satisfy the profit constraint, and diamond nodes satisfy both constraints (i.e., these are solution nodes). As is evident from the graph, the nodes that satisfy the capacity constraint are towards the left whereas the nodes that satisfy the profit constraint are towards the right. Note that, in general, as the the capacity constraint is relaxed, the number of nodes that satisfy the constraint increases proportional to $\sum_{i=0}^{k} \binom{n}{i}$, where n is the total number of items, and k is proportional to how "relaxed" the constraint is. Similar reasoning applies to the profit constraint.

Informally, when both constraints are too "tight," no solution nodes will exist, as in the case in the example displayed in Figure 1a. When the constraints are relaxed, the number of solution nodes will tend to increase and the answer to the decision problem of the instance will be yes. For example, Figure 1b shows the search space when both capacity and profit constraint are equal to 12.

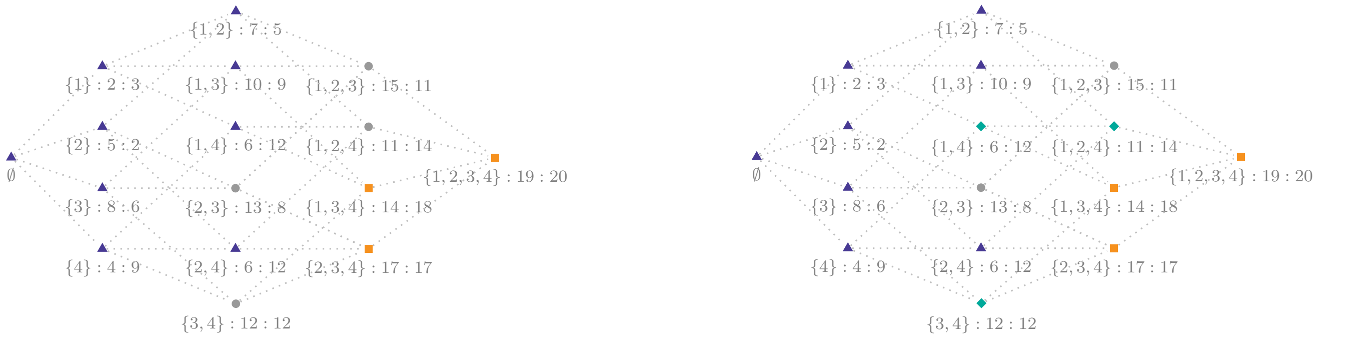
In the following, we investigate whether solvability of instances of the KP exhibits a phase transition, that is, whether there exists a region in which the the probability that both constraints can be satisfied changes precipitously from near 1 to near 0.

3 Probabilistic analysis

In this section, we characterize a phase transition in solvability for random instances of the KP. Importantly, we also show a "convexity" property of the phase transition region. Our analysis utilizes a fast greedy-type estimation rule to provide a lower bound of the probability that a given instance has a solution.

Existence of a Phase Transition. We consider a distribution of instances of the KP with varying item weights and values, keeping the number of items fixed at n. Intuitively, a phase transition emerges around critical values of capacity and profit for which the probability that a random instance has a solution changes from zero to one.

To achieve meaningful comparisons across instances, we normalize capacity and profit as follows. Given a tuple of positive integers $\langle \hat{x}_1, \ldots, \hat{x}_n \rangle$ and a real number $\hat{y} \in [0, \sum_{i=1}^{n} \hat{x}_i]$, we shall use the function $\sigma(\hat{y}, \langle \hat{x}_1, \ldots, \hat{x}_n \rangle) = \frac{\hat{y}}{\sum_{i=1}^{n} \hat{x}_i}$ as the *normalized value* of $\hat{y}$ with respect to $\langle \hat{x}_1, \ldots, \hat{x}_n \rangle$. Note that the range of the function σ is $[0, 1]$.



(a) Weight and profit constraints of 10 and 15, resp.

(b) Weight and profit constraints of 12 and 12, resp.

Figure 1: Search space of a knapsack instance with 4 items with weights $\{2, 5, 8, 4\}$ and values $\{3, 2, 6, 9\}$. Nodes that satisfy the weight constraint are indicated by a triangle, nodes that satisfy the profit constraint are indicated by a square, and nodes that satisfy both constrains are indicated by a diamond.

Thus, given a knapsack instance with items $I = \{1, \ldots, n\}$ having weights $\langle \hat{w}_1, \ldots, \hat{w}_n \rangle$, values $\langle \hat{v}_1, \ldots, \hat{v}_n \rangle$, capacity constraint $\hat{c}$ and profit constraint $\hat{p}$, we denote the *normalized capacity constraint* by $c = \sigma(\hat{c}, \langle \hat{w}_1, \ldots, \hat{w}_n \rangle)$ and the *normalized profit constraint* by $p = \sigma(\hat{p}, \langle \hat{v}_1, \ldots, \hat{v}_n \rangle)$. A set of items $S \subseteq I$ is a *solution* to an instance when *(a)* the normalized sum of item weights in S is at most the normalized capacity, that is, $\sigma(\sum_{i \in S} \hat{w}_i, \langle \hat{w}_1, \ldots, \hat{w}_n \rangle) \leq c$; and *(b)* the normalized sum of item values in S is at least the normalized profit, that is, $\sigma(\sum_{i \in S} \hat{v}_i, \langle \hat{v}_1, \ldots, \hat{v}_n \rangle) \geq p$. *In the following we use this normalized version as our canonical definition of a knapsack instance* and look at phase transitions across two measures: *(i)* the space (c, p) of normalized capacity and profit values; and *(ii)* the space $r = c/p$ of the ratio between normalized capacity and profit.[5]

We do not impose any restrictions on the distribution from which values and weights are drawn, except that the values are continuous, independent and identically distributed. An example would be gamma-distributed values and weights, in which case the normalized values and weights follow a Dirichlet distribution [1].

Regarding the capacity constraint, we shall consider c (normalized) values within $[0, 1]$. Regarding the profit constraint, in turn, we look for a (normalized) p that can always be met by any sampled instance. Note that the ratio c/p does not exist for $p = 0$. Therefore, we will only consider values of p within $[p_{\min}, 1]$, where $p_{\min} = 1/n$. An item with a normalized value at least $1/n$ is guaranteed to always exist in any sampled instance (because of the restriction that all values are drawn from identical distributions and that normalized item values add up to 1). As a result, the ratio $r = c/p \in [0, n]$.

We now aim to better understand the boundaries of the phase transition region. We do so by studying the event $E(c, p)$, the collection of weights and values for all n items for which the resulting KP instances (with capacity c and profit p) admit a solution.

Lower bounds for $P[E(c, p)]$. We first consider a weak bound. Given a KP instance, we order the items arbitrarily, thus creating arbitrary sequences of values and weights that (after normalization) add up to 1. We then consider knapsacks with increasing number of items $s \geq 1$, which we fill in the order of the sequence. For the knapsack with the first s items, we determine whether the capacity and the profit constraints are met. The process is similar to executing an incomplete algorithm that at each step adds the "next" item to the knapsack and checks if the capacity and profit constraints are satisfied. In the worst case, such an algorithm would execute n such steps.

We consider the event $E^l(c, p)$ as a collection of weight and value assignments for the items, together with *an ordering on those items* such that for each assignment there is only one ordering in the event set, and such that the capacity and profit constraints c and p, resp., are met by taking the first $s \geq 1$ items in the corresponding ordering. It follows then that $P[E^l(c, p)]$ provides a *lower bound* for $P[E(c, p)]$ since, under E^l, we only consider one possible ordering for each instance (a re-ordering of the items may make the capacity and profit constraints hold when the original, random ordering, did not). Formally, $P[E^l(c, p)] \leq P[E(c, p)]$, for all c and p.

To get a closed-form expression of $P[E^l(c, p)]$, we sum the probability of a solution over the possible sizes of the knapsack (i.e., from 0 to n). The probability that a knapsack of size k is a solution is equal to the product of two probabilities: (i) the probability that the capacity constraint is satisfied *exactly* at k (a partition on the capacity constraint) and, (ii) the probability that the profit constraint is satisfied. Now, we can write $P[E^l(c, p)]$ more explicitly as follows (where the constraint $\sum_{i=1}^{n+1} w_i > c$ is assumed to be always true):

$$P[E^l(c, p)] = \sum_{s=0}^{n} f(s) \times G(s), \tag{1}$$

where

- $f(s) = P[\sum_{i=1}^{s} w_i \leq c \wedge \sum_{i=1}^{s+1} w_i > c]$ denotes the probability (density) that the knapsack reaches the (maximum feasible) capacity at s; and
- $G(s) = P[\sum_{i=1}^{s} v_i \geq p]$ denotes the *cumulative* probability that the knapsack reaches the profit constraint with the first s items.

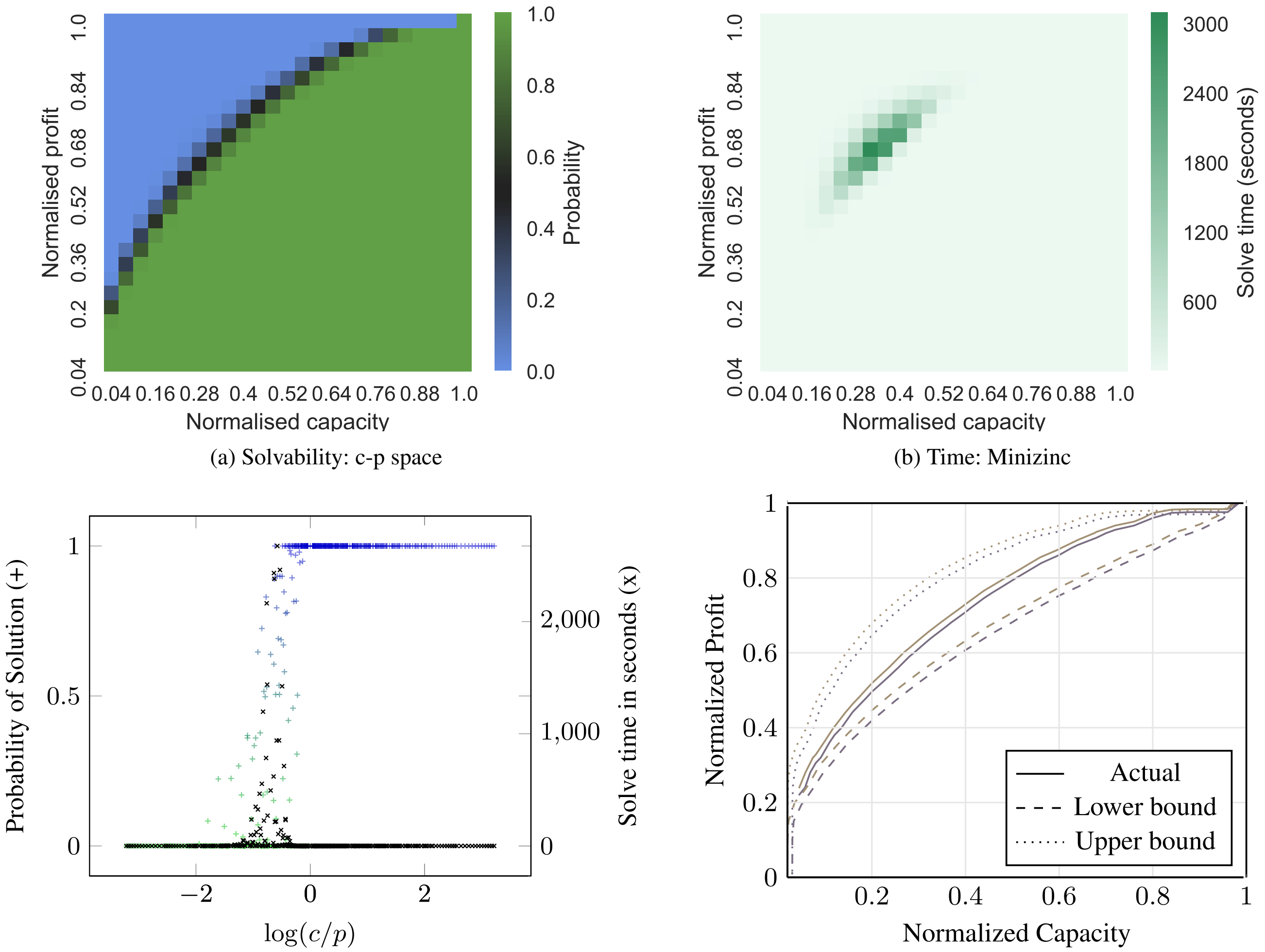
Integrating Equation 1, we then obtain:

$$P[E^l(c, p)] = 1 - \sum_{0}^{n} F(s) g(s),$$

where $F(s) = \sum_{0}^{s} f(s)$ and $g(s) = G(s) - G(s-1)$.

We now consider a *stronger* bound by re-arranging the sequence of items so that they are in *ascending order* of weights. We refer to this strategy as *weight-greedy*, which does not affect $G(s)$ (or $g(s)$) since values are drawn independently. However, $\tilde{F}(s)$, the (cumulative) probability that the knapsack reaches capacity at $t \leq s$, is now smaller than $F(s)$, with strict inequality for some s. Indeed, if at s items, capacity is reached with the unordered sequence, there is a

[5] We denote the normalized value of a parameter $\hat{x}$ by x.



(a) Solvability: c-p space
(b) Time: Minizinc
(c) Solvability: c/p space
(d) Isoquants: Actual vs Upper and Lower bounds.

Figure 2: Phase transition and time complexity for knapsack instances with 50 items, weights/values sampled from a uniform distribution from a range of $[0, 10^7]$. The isoquant contour pairs are at 40% and 60%.

positive chance that items in position $t, t > s$, are smaller (since the weights have to add up to 1), and hence, these would come earlier in the ordered sequence, implying that capacity is reached only at a larger s. Let $\tilde{f}(s) = \tilde{F}(s) - \tilde{F}(s-1)$.

Let $E^L(c, p)$ denote the event that a knapsack instance has a solution at c and p when the items are re-ordered according to increasing weight. Then, following a similar derivation to Equation 1 we get:

$$P[E^L(c,p)] = \sum_0^n \tilde{f}(s)G(s) = 1 - \sum_0^n \tilde{F}(s)g(s).$$

Since $\tilde{F}(s) < F(s)$, we have that $P[E^L(c,p)]$ is a better lower bound for $P[E(c,p)]$ (i.e., $P[E^L(c,p)] > P[E^l(c,p)]$ and $P[E^L(c,p)] \leq P[E(c,p)]$).

We now show that the *location of the phase transition region is above the 45 degree line in* (c, p) *space (or below* 1 *in* r *space); and that it exhibits a 'convex' shape*. If $\tilde{F}$ and g are non-negatively correlated as a function of s, we can formally express the location of the isoquant where the lower bound equals 1 as follows:

$$\sum_0^n \tilde{F}(s)g(s) \geq n \left(\frac{1}{n} \sum_0^n \tilde{F}(s) \frac{1}{n} \sum_0^n g(s) \right) > 1/n.$$

Hence, $P[E^L(c,p)] = 1 - \sum_0^n \tilde{F}(s)g(s) < 1 - 1/n$, or $P[E^L(c,p)] < 1$ for any c, p where $\tilde{F}$ and g are non-negatively correlated. A non-negative correlation between $\tilde{F}$ and g is most likely to occur for $c < p$: $\tilde{F}$ is monotone increasing towards 1, which it reaches for low $s = s^*$ (s^* is the value at which the knapsack is filled to capacity), after which it is flat; $g(s)$ is increasing in s and will only decrease beyond s^* since $p > c$. Note that, even if $c < p$, for c close to p, and for small values of c, $\tilde{F}$ reaches 1 only at higher s^* since the items are ordered by weight. At the same time, the profit constraint may be satisfied for low values of s, and $g(s)$ peaks before reaching s^*. The correlation constraints hold for most of the (c, p) space when c is sufficiently high, which implies that the lower bound of the contour of probabilities where $P[E^L(c,p)] = 1$ should be below the 45 degree line for high c. In r space, the phase transition therefore stops at a value r_1 which is close to or below 1.

Convexity. Now consider the case where $c = p$ (i.e., $r = 1$). Notice that, by symmetry (weights and values have the same distribution):

$$P[E^l(c,c)] = \sum_0^n f(s)F(s) = 1 - \sum_0^n F(s)f(s),$$

so $P[E^l(c,c)] = 1/2$. Hence, it follows that $1/2 < P[E^L(c,c)] < P[E(c,c)](= P[E(1)])$. That is, *if* $c = p$ *(i.e.,* $r = 1$*), the probability that a solution exists is strictly larger than* $1/2$. As a result, *maximum uncertainty (entropy) about the existence of a solution occurs in the region* $c < p$*, i.e., above the diagonal in* (c, p) *space*. We refer to this as *convexity of the phase transition region*.

Upper bounds for $P[E(c,p)]$. Using analogous arguments, we can obtain an *upper* bound of the probability that a solution of a KP instance exists. Consider a *double-greedy* strategy whereby one first re-arranges the items according to ascending weight, and then re-assigns the values to the items in descending order. Let P be a KP instance with items $I = \{1, \ldots, n\}$, weights $\langle w_1, \ldots, w_n\rangle$, values $\langle v_1, \ldots, v_n\rangle$, a capacity c and profit threshold p. We assume that items I are arranged in ascending order of weights (this does not change the KP instance). The double-greedy algorithm manipulates the values of the items. We denote these values by $\{v'_1, \ldots, v'_n\}$. The double-greedy algorithm first constructs a knapsack by adding items incrementally until the capacity constraint is reached. Then, it checks if the value of this knapsack reaches the profit threshold. Formally, the algorithm constructs a knapsack $\{1, \ldots, k-1\}$ such that $\sum_{i<k} w_i \leq c$ and $\sum_{i \leq k} w_i + w_k > c$. Then, if $\sum_{i<k} v'_i \geq p$, the algorithm terminates and outputs the set of items as a solution; otherwise, it terminates without any output.

We now show that this double-sided algorithm is complete (but not sound). That is, if the given instance has a solution then it will also generate a solution (albeit an incorrect one). However, the double-greedy strategy may also generate a solution for unsolvable instances. We begin by pointing out that since the values of items have been re-assigned in descending order, the sum of re-assigned values of the first k items will be at least the sum of values of any set of k items. That is, $\sum_{i \leq k} v'_i \geq \sum_{i \in A} v_i$ where A is a subset of I of size k. Analogously, the following property also holds for the weights, that is, $\sum_{i \leq k} w_i \leq \sum_{i \in A} w_i$, where A is a subset of I of size k.

Suppose that instance P has a solution. That is, there exists a subset $A^* \subseteq I$ containing $k^* \leq n$ items such that $\sum_{i \in A} w_i \leq c$ and $\sum_{i \in A} v_i \geq p$. Hence, $\sum_{i \leq k^*} w_i \leq \sum_{i \in A} w_i \leq c$ and $\sum_{i \leq k^*} v'_i \geq \sum_{i \in A} v_i \geq p$. Therefore, the double-greedy algorithm will output the set $\{1, \ldots, k^*\}$ as a solution.

The probability that the double-sided greedy algorithm produces a "solution" is at least as high as the probability of a KP instance having a solution. Therefore, it provides an upper bound.

4 Computational experiments

We now examine the relation between the time complexity of solving random KP instances and the phase transition in solvability empirically. We also analyse the relation between the actual frequencies of solvability and the theoretical lower bounds. We will consider random instances with $n \in \{20, 30, 40, 50\}$ items. For each n, we sampled (with replacement) a collection of weight and value combinations (a combination is $(\langle \hat{w}_1, \ldots, \hat{w}_n\rangle, \langle \hat{v}_1, \ldots, \hat{v}_n\rangle)$). Weights and values were always sampled from the same discrete uniform distribution with range $\{0, \ldots, 10^7\}$. In the second step, for each combination, multiple knapsack instances were generated by considering normalized capacity and normalized profit constraints at regular intervals of 0.04 in $[0, 1]$, respectively.

To solve the instances, we used two existing generic off-the-shelf solvers that are based on different solving methods, *Minizinc* [21] and *Minisat+* [5]. Minizinc is a constraint solver that uses a branch-and-bound technique with constraint propagation. Minisat+, on the other hand, is a pseudo-boolean satisfiability solver (based on Minisat) that uses conflict resolution [4]. To obtain an unbiased and optimal constraint solver model, we used the global "knapsack" constraint of the Minizinc library. We used these two different solvers to reduce the possibility that our results are biased by a particular solution technique. The results of the computational experiments for instances with 50 items are shown in Figure 2 (results for instances with different numbers of items were qualitatively the same).

Solvability First, we examine whether the probability of solvability of instances changes as a function of normalized capacity c and normalized profit p and, in particular, whether this probability exhibits a phase transition. The plot in Figure 2a shows the probability that an instance has a solution at given levels of normalised capacity c and normalised profit p. As is evident from the plot, the probability that an instance has a solution tends to increase in c and decrease in p, ceteris paribus, as expected. In other words, solvable instances generally have relaxed constraints (i.e., normalised capacity is higher than normalised profit) while unsolvable instances have tight constraints (i.e., normalised capacity is lower than normalised profit).

In a contour plot of the probability of a solution as a function of c and p with equi-distant contours, the probability is *above* the 45 degree line (see Figure 2d). Likewise, in a plot of $P[E(r)]$ as a function of the log-ratio $\log(r) = \log(c/p)$, the region of maximum uncertainty—where the probability of a solution is $1/2$—is to the *left* of 0 (see Figure 2c).

Time complexity Next, we examine time complexity of the instances. We computed the time taken to solve the random instances by Minizinc (Figure 2b and 2c). The plots show that the hard instances occur around the phase transition from solvability to unsolvability (region of maximum entropy in the probability space). The average compute time using Minisat+ (not shown here) has a similar relationship to the phase transition as for Minizinc. Hard instances for both solvers were located in the same area of the parameter space, indicating the robustness of our findings with regards to the solution technique. Also observe that we are able to identify an island of difficulty within the instances that lies in the phase transition (see Figure 2b). The hardest instances occur between values of normalised capacity of 0.4 and 0.52 and values of normalised profit of 0.52 and 0.68.

Phase transition To characterise the phase transition in the KP in terms of c and p, we plot the probability of solvability as a function of c and p (Figure 2a). We also use the log of this ratio (i.e., $\log(c/p)$) to plot the solvability as a function of a single metric (Figure 2c). Note that similarly to phase transitions in solvability observed in relation to other NP-complete problems such as 3-SAT, integer partition and graph colouring, the phase transition from solvability to unsolvability in the KP occurs precipitously near 0, that is, near the point $c = p$.

Importantly, the probability that an instance has a solution, has a

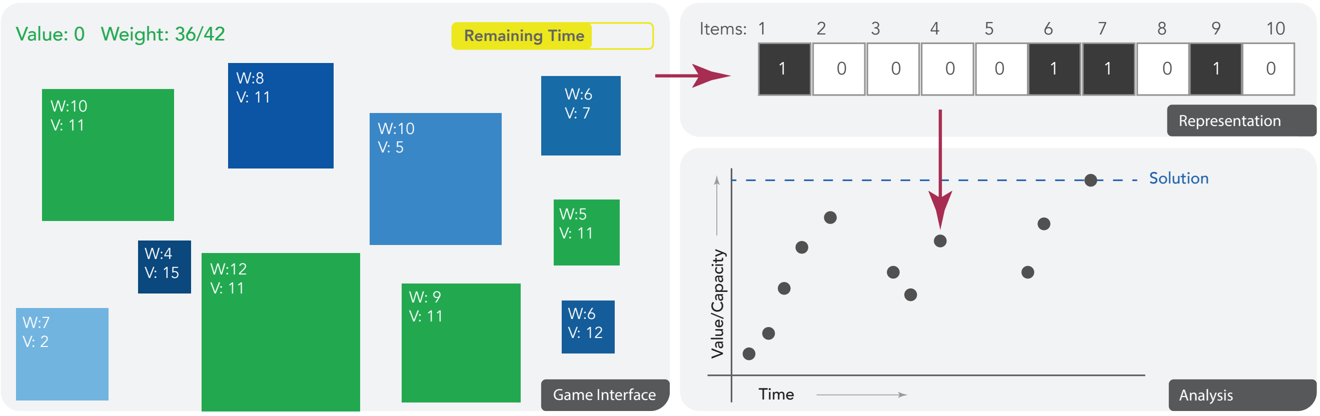


Figure 3: Experimental paradigm: Example instance of the 0–1 knapsack problem with 10 items with a capacity constraint of 42. In the experiment, each item was represented by a square. Selected items were displayed in green. The size of an item was proportional to its weight and the colour was proportional to its value. The total value and the total weight of the current knapsack were displayed at the top of the screen. The internal representation of a state of the knapsack was represented by an array of 0 s and 1 s with length equal to the number of items available. To analyse the time spent we compared the ratio of value and capacity of a knapsack state and the time spent on that state.

phase transition above the 45 degree line in (c,p) space. Our computational results also show that the transition from solvability to unsolvability exhibits a *convex* shape. In the area near the centre of the diagonal where $c = p$, instances with $c < p$ have a higher probability of having a solution at the centre of the diagonal than instances at the ends of the diagonal. This finding confirms our result in Section 3.

This convexity is related to structural properties of the search space. Intuitively, the number of possible knapsack configurations is maximal when the size of the knapsack is half the total number of items (the function $\binom{n}{k}$ peaks at $k = n/2$). In addition, given that we are considering random instances, there is a probability that there are some items whose weights are (relatively) smaller than their values. Therefore, the actual probability of finding a knapsack configuration that is a solution to an instance with lower normalised capacity than normalised profit, peaks when the knapsack configuration size is half of the total number of items.

Using simulations, we now investigate how tight the upper and lower bounds are. We randomly generated 1,000 knapsack instances with $n = 50$, normalized weights and values, applied the greedy strategies and determined, for each combination of c and p (in intervals of 0.05, from 0 to 1), whether there exists a solution. Figure 2d plots the isoquants in (c,p) space. In (c,p) space, the phase transition region clearly shows convexity; virtually all (equi-distant) isoquants are above the 45 degree line. The region exhibits the same features as the one obtained from the true (estimated) probabilities.

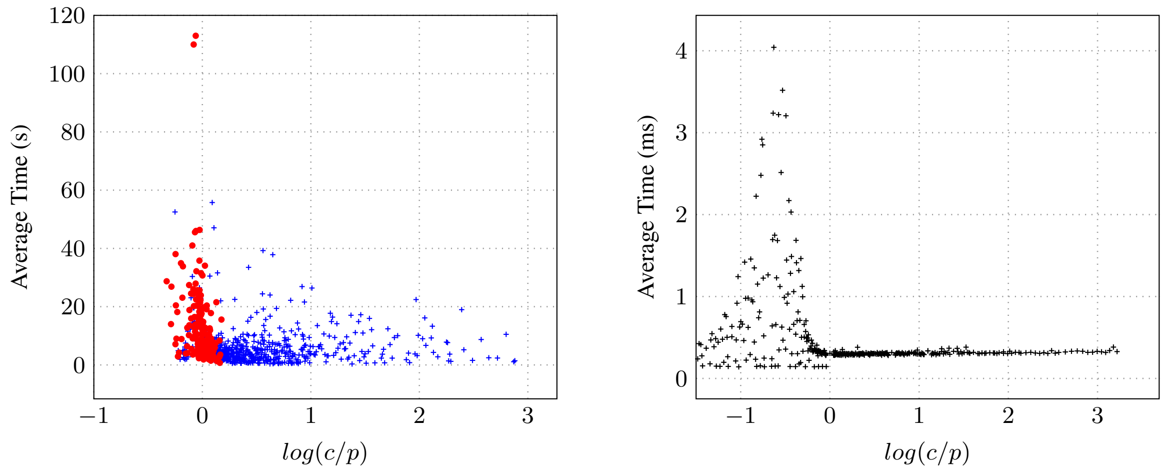
5 Human Experiments

As argued before, the KP is ubiquitous in every-day human life, at many different levels of cognition [26]. An interesting question, therefore, is whether instances near the phase transition are also harder for humans. At the surface, there are many inherent differences between electronic and human computers [20]. For example, as compared to electronic computers, humans are more memory-constrained and therefore have limited capacity to implement solution techniques such as dynamic programming. In addition, humans are affected by relatively short attention spans, fatigue, and calculation errors. On the other hand, recent experimental evidence suggests that instance complexity does predict human behaviour. It has been shown that as instance complexity of the optimisation variant of the KP increases, the probability of human participants being able to solve the instance decreases [16, 20]. However, these earlier studies used algorithm-dependent metrics of instance complexity.

In the following, we examine whether instances near the phase transition identified above are also harder for humans. To investigate this question, we used data from an experimental study where human participants were asked to solve a number of instances of the optimisation variant of the 0-1 knapsack problem (Figure 3 shows a schematic view of the experimental paradigm). Twenty-two human volunteers (age range = 18-30, mean age = 22.2, 17 female, 5 male) recruited from the general population took part in the study. The experimental protocol was approved by the University of Melbourne's human research ethics committee. The order of instances was randomised in each session. Participants were incentivized as follows: (i) for each instance they received a cash amount proportional to the total value of their knapsack (relative to the value of the optimal knapsack); and (ii) a fixed show-up fee. The experiment included a training session prior to actual testing. The optimisation variant of the KP can be considered as a sequence of decision tasks in which participants have to answer the question "Does there exist another set of items with a higher profit than the current set?" We take the time spent at each node as a proxy of the time participants spent on solving the corresponding decision problem. In the experiment, participants were asked to find the set of items with the highest total value subject to the capacity constraint. They always started with an empty knapsack and had 240 seconds to solve an instance. They used a computer interface to add items to and remove items from the the knapsack. Each participant solved 15 instances. Each instance had 10 items. Item values and weights, as well as capacity, differed across instances. Values and weights were at most three digits long and were drawn from the same distribution.

We call each addition of an item to or removal from the knapsack a *move*. We measured the time taken between each move. We then calculated normalised capacity c and normalised profit p for each feasible search state that participants visited (participants could not reach infeasible search states).

The time spent by participants in a given state is a function of two key cognitive processes: (i) parsing of information (profit and



(a) Average time spent on nodes.

(b) Average time by Minizinc

Figure 4: Human complexity of solving Knapsack instances.

weight of the state); and (ii) solving of instance of the decision problem (does there exist a feasible state with a higher total profit?). At the start of each instance, a participant will consume more time to parse information (e.g., weight and value combinations of items) and therefore the time absorbed by this process will be higher. After having parsed the information, the time taken to solve an instance would be the main driver of the time spent in a state.

Figure 4 compares the average time spent by human participants and by the Minizinc solver (on instances with 20 items) as a function of $log(c/p)$. The nodes colored in red are the nodes corresponding to local maxima in the search space (i.e., states in which no additional item could be added without violating the capacity constraint). As Figure 4a shows, the average time taken is maximal to the left of the phase transition (i.e., just before $log(c/p) = 0$). As is evident from Figure 4, time increases the closer a search node is to the phase transition boundary. This finding provides empirical evidence that instances near the phase transition were also harder for humans. We note that we have successfully reproduced these findings in repetitions of the same experimental paradigm.

6 Discussion

Key contributions in AI have emerged as a result of research on human intelligence (e.g., development of heuristics and algorithms). Here, we go the other way and use knowledge about phase transitions developed in AI to study human reasoning. We argue that our work provides a step towards establishing common ground for interdisciplinary collaborations between human decision making and artificial intelligence. This will allow an understanding of tasks for which humans are good at computing solutions and tasks where humans may need inputs from machines for computing solutions.

Given the ubiquity of the KP in every-day life, a better characterisation of its structure can improve our understanding of human computation and behavior, including the use and development of heuristics and the occurrence of biases. In addition, identifying properties which impact the accuracy of human computation will help in devising better mechanisms for human-computer interactions.

Problems in class NP-complete are considered intractable if, in the worst case, the time taken by a computer to solve them grows faster than polynomially in input size. However, many instances of NP-complete problems can nevertheless be solved in polynomial time. Understanding what the hard instances are and how they affect human decision making, is an important question.

It has been conjectured that all NP-complete problems have at least one order parameter such that the phase transition in solvability occurs near a critical value of this parameter [3]. NP-complete problems for which such parameters have been identified include satisfiability [17, 9], integer partition [11], graph colouring [3], and traveling salesman [10]. In this paper, we identified such a parameter for the 0-1 KP, providing further evidence towards the conjecture.

A generalization of this work conjectures that phase transitions in solvability are related to the constrainedness of instances [8]. Constrainedness of search over an ensemble can be estimated by the formula $\kappa = 1 - \frac{log_2(\mathbb{E}[Sol])}{N}$, where $N = log_2(|S|)$ (i.e., log of the size of the total search space S), and $\mathbb{E}[Sol]$ is the expected number of solutions. Instances with $\kappa < 1$ are considered under-constrained while instances with $\kappa > 1$ are over-constrained. Phase transitions have been shown to exist (e.g., for 3-SAT and graph colouring) when $\kappa \approx 1$. We showed that for the KP (under the sampling assumptions of Section 3), $\kappa \approx 1$ when $c/p \approx 1$, where c and p are normalized capacity and normalised profit, respectively.

In this paper, we also provide empirical evidence that difficulty for human decision-makers increases close to the phase transition. Although there are other NP-Complete problems (e.g., the widely-studied 3-SAT) for which phase transitions have been identified, we believe that the KP is more suited to study the role of complexity in human decision making. This is partly because of the nature of the task (logical in 3-SAT vs search/optimisation in the KP), representational issues (with just 5 variables the number of clauses to be displayed will be 25 for a c/l ratio of 5), and a cleaner experimental task design (the displayed input size remains the same across KP instances, whereas in 3-SAT the number of clauses varies, and non-computer science participants tend to understand a task based on the KP more easily).

Recent evidence [20, 2] suggests that, due to limited working memory, humans may not resort to dynamic programming to compute the optimal value of a KP instance. Evidence presented in this paper provides further support of this conjecture. In theory, one can solve the decision case by first solving a corresponding optimisation variant of the same instance and then checking the optimal value of the knapsack against the given profit threshold. However, this is ineffective computationally. If the profit threshold is low as compared to the capacity threshold, a better algorithm would be able to solve that instance with less effort. In terms of the search space, nodes with low profit and high capacity would be outside the phase transition region. In our experiments, participants spent more time on nodes that were closer to the phase transition region as compared to nodes that were further away from the phase transition.

In future work, we will investigate how human reasoning is affected by phase transitions in solvability in other NP-complete problems, and the relation between decision and optimisation variants. In addition, we aim to design experiments to better understand the limits to human computation.

REFERENCES

[1] Martin Bilodeau and David Brenner, *Theory of multivariate statistics*, Springer Science & Business Media, 1999.

[2] Peter Bossaerts and Carsten Murawski, 'Computational complexity and human decision-making', *Trends in Cognitive Sciences*, **21**(12), 917–929, (2017).

[3] Peter Cheeseman, Bob Kanefsky, and William M Taylor, 'Where the Really Hard Problems Are', *The 12nd International Joint Conference on Artificial Intelligence*, 331–337, (1991).

[4] Martin Davis, George Logemann, and Donald Loveland, 'A machine program for theorem-proving', *Communications of the ACM*, **5**(7), 394–397, (1962).

[5] Niklas Eén and Niklas Sorensson, 'Translating pseudo-boolean constraints into sat', *Journal on Satisfiability, Boolean Modeling and Computation*, **2**, 1–26, (2006).

[6] Michael R Garey and David S Johnson, "'strong"np-completeness results: Motivation, examples, and implications', *Journal of the ACM (JACM)*, **25**(3), 499–508, (1978).

[7] Michael R. Garey and David S. Johnson, *A Guide to the Theory of NP-Completeness*, W. H. Freeman & Co., 1979.

[8] Ian P Gent, Ewan MacIntyre, Patrick Prosser, Toby Walsh, et al., 'The constrainedness of search', in *AAAI/IAAI, Vol. 1*, pp. 246–252, (1996).

[9] Ian P. Gent and Toby Walsh, 'Easy problems are sometimes hard', *Artificial Intelligence*, **70**(1-2), 335–345, (1994).

[10] Ian P Gent and Toby Walsh, 'The TSP phase transition', *Artificial Intelligence*, **88**(1-2), 349–358, (1996).

[11] Ian P. Gent and Toby Walsh, 'Analysis of Heuristics for Number Partitioning', *Computational Intelligence*, **14**(3), 430–451, (1998).

[12] *Frontiers in Problem Solving: Phase Transitions and Complexity*, eds., T. Hogg, B. A. Huberman, and C. Williams, volume 81 of *1–2*, Elsevier Science Publishers Ltd., 1996. Special issue of Artificial Intelligence.

[13] Hans Kellerer, Ulrich Pferschy, and David Pisinger, *Knapsack Problems*, Springer, 2004.

[14] Renata Mansini and Maria Grazia Speranza, 'Heuristic algorithms for the portfolio selection problem with minimum transaction lots', *European Journal of Operational Research*, **114**(2), 219–233, (1999).

[15] G. B. Mathews, 'On the partition of numbers', *Proceedings of the London Mathematical Society*, **28**, 486–490, (1897).

[16] Debrah Meloso, Jernej Copic, and Peter Bossaerts, 'Promoting intellectual discovery: patents versus markets.', *Science*, **323**(5919), 1335–1339, (March 2009).

[17] David G. Mitchell, Bart Selman, and Hector J. Levesque, 'Hard and easy distributions of SAT problems', in *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp. 459–465, (1992).

[18] Remi Monasson, Riccardo Zecchina, Scott Kirkpatrick, Bart Selman, and Lidror Troyansky, 'Determining computational complexity from characteristic 'phase transitions'', *Nature*, **400**(6740), 133–137, (July 1999).

[19] Cristopher Moore and Stephan Mertens, *The nature of computation*, OUP Oxford, 2011.

[20] Carsten Murawski and Peter Bossaerts, 'How humans solve complex problems: The case of the knapsack problem', *Scientific Reports*, **6**, (10 2016).

[21] Nicholas Nethercote, Peter J Stuckey, Ralph Becket, Sebastian Brand, Gregory J Duck, and Guido Tack, 'Minizinc: Towards a standard cp modelling language', in *International Conference on Principles and Practice of Constraint Programming*, pp. 529–543. Springer, (2007).

[22] John von Neumann, *The Computer and the Brain*, Yale University Press, New Haven, CT, USA, 1958.

[23] David Pisinger, 'Where are the hard knapsack problems?', *Journal of Computers and Operations Research*, **32**(9), 2271–2284, (2005).

[24] Sartaj Sahni, 'Approximate algorithms for the 0/1 knapsack problem', *Journal of the ACM (JACM)*, **22**(1), 115–124, (1975).

[25] Kate Smith-Miles and Leo Lopes, 'Measuring instance difficulty for combinatorial optimization problems', *Computers & Operations Research*, **39**(5), 875–889, (2012).

[26] Teresa Torralva, Ezequiel Gleichgerrcht, Alicia Lischinsky, María Roca, and Facundo Manes, '"ecological" and highly demanding executive tasks detect real-life deficits in high-functioning adult adhd patients', *Journal of Attention Disorders*, **17**(1), 11–19, (2013).