Computational complexity of decisions:

Quantifying computational hardness and its effects on human computation

Juan Pablo Franco

ORCID Identifier: 0000-0003-2608-2035

Doctor of Philosophy

October 2021

Faculty of Business and Economics

THE UNIVERSITY OF MELBOURNE

Submitted in total fulfillment for the degree of
Doctor of Philosophy

Abstract

Humans are presented daily with decisions that require solving complex problems. In many cases, solving these problems is computationally hard. This raises a tension between the computational capacity of the agent and the computational requirements of a task. Whilst the underlying invariants of this mechanism remain unclear in cognition, they have been widely studied in computer science. I build on theoretical and empirical work in computational complexity, which characterizes the intrinsic computational hardness of problems. I first present an adaptation of this theoretical framework for the study of human cognition by introducing a set of metrics of hardness of instances of problems. I do this in a way that is independent of any algorithm or computational model and that can be generalized to other problems. Based on this, I explore empirically, in a set of lab experiments, how these task-independent metrics of hardness affect human problem-solving. I do this at two levels of analysis. Firstly, I study how these metrics affect human performance at the behavioral level in three canonical computational problems: the knapsack problem, the traveling salesperson problem and the Boolean satisfiability problem. Secondly, I examine the relation between computational hardness and the neural processes associated with problem-solving, employing ultra-high field functional MRI. I find that the metrics of intrinsic hardness put forward here predict human performance and time-on-task across the three computational problems in a similar way. Moreover, I identify the neural correlates of computational hardness in the knapsack task, a complex problem-solving task. I show that this framework can be used for the study of the neural underpinnings of problem-solving by providing a generic definition of cognitive demand. The results of these studies provide support for the conceptual premise that the quantification of intrinsic hardness is fundamental in the development of more refined theories of human decision-making and its neural underpinnings. Critically, they provide a framework to study how humans adapt to computational complexity and how intrinsic hardness of tasks affect the reliability of human decision-making. This could inform public policy by identifying which decisions over products involve solving problems that require computational resources beyond those available to an agent, and how this affects decisions.

Declaration

This is to certify that:

i. The thesis comprises only my original work towards the PhD except where indicated in the Preface.

ii. Due acknowledgment has been made in the text to all other material used.

iii. The thesis is fewer than the maximum word limit (100,000 words) in length, exclusive of tables, maps, bibliographies and appendices.

Signed:

Juan Pablo Franco

Preface

The work presented here is interdisciplinary, encompassing insights from computer science, cognitive sciences and research in neuroscience. Presentation of this work to an audience that might span several fields generates a continuous tension between depth and clarity. Therefore, I present this manuscript in a way that resembles my own discovery path during my PhD. Explicitly, I present the introduction in a way that assumes no previous knowledge of computing theory. I hope the informed reader will not regard this as sign of arrogance, and instead skip ahead when pertinent. The co-authored papers presented in chapters 3, 4 and 5 are written for different audiences (see below) and will thus reflect this.

I would also like to highlight that although in chapters 1, 2 and 6 I use the pronoun "I" to describe the work in this thesis, the conception of the framework and metrics proposed here are the product of an insightful and continuous dialogue with my supervisors and my co-authors.

Contribution declaration

This thesis contains original research in Chapters 2 through 5. Chapters 3, 4 and 5 are based on multi-authored articles.

Chapter 3

Juan Pablo Franco, Nitin Yadav, Peter Bossaerts, Carsten Murawski
Generic properties of a computational task predict human effort and performance
In revision following peer review by Journal of Mathematical Psychology.

Author contributions:

- JPF contributed 80% of the content of the article.
- CM, JPF, PB and NY designed the study; NY and JPF performed sampling of instances; JPF programmed the experimental tasks; JPF performed data collection and analysis; JPF wrote the first draft of the manuscript; CM, JPF, NY and PB all contributed to the writing and editing of the current version of the manuscript.

Chapter 4

Juan Pablo Franco, Karlo Doroc, Nitin Yadav, Peter Bossaerts, Carsten Murawski
Task-independent metrics of computational hardness predict performance

of human problem-solving
Submitted for publication to Science on April 2021.

Author contributions:

- JPF contributed 70% of the content of the article.
- CM, JPF, KD, PB and NY designed the study; NY and JPF performed sampling of instances; KD and JPF programmed the experimental tasks; KD ran a pilot version of this study; JPF performed data collection and analysis of the main study; JPF wrote the first draft of the manuscript; JPF, CM, KD, NY and PB all contributed to the writing and editing of the current version of the manuscript.

Chapter 5

Juan Pablo Franco, Peter Bossaerts, Carsten Murawski
The dynamics of neural correlates of complex problem-solving
Unpublished material not submitted for publication.

Author contributions:

- JPF contributed 80% of the content of the article.
- JPF, PB and CM designed the study; JPF programmed the experimental tasks; JPF performed data collection; JPF performed behavioral data analysis; JPF, CM and PB performed neuroimaging data analysis; JPF wrote the first draft of the manuscript; CM, JPF and PB all contributed to the writing and editing of the current version of the manuscript.

Funding sources

This PhD was funded by the University of Melbourne Graduate Research Scholarship from the Faculty of Business and Economics. Financial support was also received through a Research Development Grant from the Faculty of Business and Economics.

Acknowledgments

I feel immensely grateful for having had the opportunity to pursue these doctoral studies under the brilliant supervision of Carsten Murawski and Peter Bossaerts. I cannot begin to express my thanks for all the time and effort they have so generously offered me. Peter has been a beacon of knowledge always willing to provide insightful suggestions and constructive feedback. I am particularly indebted to my primary supervisor, Carsten, for his unparalleled generosity. I count myself lucky to benefit from his extensive knowledge and his relentless support. Not only has he provided me with invaluable feedback throughout my PhD, he has always been quick to offer his time to engage in insightful and thought-provoking discussions.

As part of my studies I had the chance to work as well with two incredible co-authors. Nitin Yadav, who was always happy to share his time and extensive knowledge about computing theory, and Karlo Doroc, with whom I spent several evenings figuring out the paradoxes of Unity.

I am also grateful to Sebastian Sardina, John O'Doherty and John Handley for making part of my advisory committee during my candidature. Your time and advice I really appreciate.

I had the privilege of undergoing my studies in an incredibly collaborative environment where everyone was always ready to lend a hand. To all members of the Brain, Mind and Markets lab: thank you! Having had the chance to work in such a great and enthusiastic team has been a great inspiration. Plus, it must be noted that, at some point or another, each of the members of the lab volunteered their time to serve as pilot subjects. I want to specially thank Elizabeth Bowman for all of her support with the laboratory experiments.

I would also like to gratefully acknowledge the assistance of Rebecca Glarin for her support of the neuroimaging sessions and Scott Kolbe for his guidance on the 7T MRI setup.

I thank the Department of Finance and the Faculty of Business and Economics at the University of Melbourne for the opportunity to undertake this PhD and for providing the funding that made this possible. I want to thank the Faculty for their outstanding commitment to the well-being of their PhD students and in particular to Jennifer Decolongon for her invaluable support.

Thanks should also go to the ResBaz team. Being part of such an outstanding and dedicated team of people allowed me to not only grow professionally, but also at a personal level. My PhD work would have not been the same without the innumerable lessons I learned from being part of this community.

I would also like to extend my deepest gratitude to my parents and my siblings:

> Porque este doctorado no hubiera sido posible sin el apoyo que recibí de mi familia a cada momento del camino. A mi mamá y a mi papá por su

apoyo incondicional: no estaría hoy aquí si no fuera por sus sacrificios y su inagotable poder de motivación. A Tin porque eres una inspiración (¡a pesar de que hayas ganado la carrera!) y a Du, porque tu cariño incondicional me motiva a ser una mejor persona a cada instante.

Finally, I cannot begin to express my thanks to my partner, Lee, without whom I would have lost my mind midway through this PhD (especially during COVID lock-downs). Your unwavering support and patience has been truly invaluable at so many levels. Thank you!

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Humans are presented daily with a plethora of decisions. In many cases, these involve performing cognitive tasks that are computationally demanding. This generates a tension between the demands of a task and the computational capabilities of an agent. In cases in which computational demands exceeds the agent's capacities, their ability to perform a task would likely be affected. Not only would the performance of the task be hindered by computational demands, it is also likely that humans would adapt to this constraint. To date, however, very little is known about the effect of this computational constraint on human behavior, and cognition in general.

Many, if not most, theories of decision-making assume (either implicitly or explicitly) that limits of computational capabilities are never a binding constraint during decision-making and can therefore be ignored in the theory. These theories include rational choice theory (Samuelson 1938) and game theory (Nash 1950). Although a large amount of evidence has registered deviations from this notion of rationality in the form of cognitive biases (Chernev, Böckenholt, and J. Goodman 2015; Shefrin and Statman 1985; Tversky and Kahneman 1981), several of the models proposed to account for these biases still would require agents to solve computationally expensive optimization problems. These include prospect theory (Kahneman and Tversky 1979; Tversky and Kahneman 1992) and social utility models (Fehr and Schmidt 1999).

In order to incorporate the difficulty of a task into a theory, models have been constructed that incorporate computational costs into models assuming different notions of rationality (Griffiths, Lieder, and N. D. Goodman 2015; Lewis, Howes, and Singh 2014; Gershman, Horvitz, and Tenenbaum 2015). In one approach, the computational costs of a problem are incorporated as an attribute to be considered when making a decision. These models incorporate additional parameters into already computationally demanding models without analyzing the plausibility of the required computations.

Overall, these models propose a theory at Marr's computational-level of analysis[1] by characterizing a task as a mathematical problem. The assumption here is that agents are able to solve these mathematical problems "reliably", and that their behavior can be successfully captured by the proposed computational-level model[2].

[1]Marr's computational-level refers to the specification of the goal of the computation, that is, the problem that needs to be solved. I refer the reader to McClamrock 1991 for a description of Marr's levels embedded in an insightful critical discussion of the limits of this categorization.

[2]The notion of reliability here alludes to the 'as-if' assumption by which these models are supported. For a thorough dissection of this ambiguous assumption, together with the relevant

The veracity of this assumption, however, is far from obvious. Indeed several of the problems implied by these theories have been shown to be intractable (Frixione 2001; van Rooij, Blokpoel, et al. 2019). Therefore, these computational models might not resemble human behavior because in some cases the computational requirements of solving the mathematical problem (e.g., an optimization problem) exceeds the capacities of the agent. The question then arises on how to determine the *cases* in which this happens and how to model the effect this interaction (between computational capacities and the requirements of a task) has on behavior and cognition in general.

Alternative approaches have put forward theories that model the agent at Marr's algorithmic-level of analysis[3]. They suggest a way forward by postulating that agents implement strategies[4] that approximate the solutions of the computational problems implied by computational-level theories. However, these models, too, fail to propose a principled approach to characterizing the computational difficulty of strategies, and thus, also do not address the plausibility of implementation. These theories include the heuristic's program (Gerd Gigerenzer and Gaissmaier 2011), including the adaptive toolbox (Gerd. Gigerenzer and Selten 2001) and ecological rationality (Todd and Gerd Gigerenzer 2012). Critically, the so-called 'heuristics' are not only poorly defined, but might not reduce the computational requirements of a task. Firstly, it is not clear how well heuristics approximate the solution to computational problems and whether an approximation guarantee affects the computational requirements of implementing a heuristic (van Rooij, Wright, et al. 2018). Secondly, the hypothetical implementation of a heuristic requires the selection of one such strategy in the first place. This implicit theoretical cornerstone of the program implies that the agent would need to solve a computational problem that might be as difficult as the original computational problem (Otworowska et al. 2018; Rich et al. 2019).

Taken together, both algorithmic-level and computational-level models of human behavior either ignore or treat rather informally the notion of computational hardness. This raises questions about the plausibility of current models of decision-making and casts doubt on the predictive power of these models. This is particularly problematic when it is considered that several day-to-day problems people face are deemed computationally hard. For instance, tasks like going to a supermarket might involve solving the knapsack problem, which is considered to be hard for computers[5]. Other examples of problems that involve computationally hard problems include computational models of vision (John K. Tsotsos 1990), of learning (Kwisthout, Wareham, and Van Rooij 2011) among many others (van Rooij, Blokpoel, et al. 2019). If the computational problems these theories suggest people solve are indeed hard, then it would be expected that computational hardness would affect human behavior and computation in ways that are currently ignored. For instance, it would

implications, I direct the reader to van Rooij, Wright, et al. 2018.

[3]Marr's algorithmic-level of analysis refers to how a computational-level theory (i.e., problem) is implemented. Specifically, it specifies the algorithm that would transform the input of the problem to the output (i.e., solution).

[4]I use this term to refer to the informal notion of computation that has also been referred to as effective procedure or effective calculation (Turing 1937). The notion of strategy will be refined in section 1.3 based on theoretical definitions.

[5]A discrete version of grocery shopping can be characterized as a computational problem that is considered intractable (NP-hard). See 1.3.

be expected that hardness would affect behavior via a negative effect on the quality of decisions. Moreover, computational hardness is presumably an attribute of the task that is considered when making a decision. An attribute that would have implications on the meta-decisions of choosing a strategy, choosing how much effort to exert as well as deciding what level of 'reliability' to aim for. Further refinements of models of decision-making would require an understanding of how computational hardness affects human behavior and computation.

Importantly, computational difficulty varies significantly from one instance of a task to the next and it is not clear what kind of computational limitations should be incorporated into decision theory. Importantly, at present, there is no overarching theoretical framework to quantify hardness of decisions in a generic way such that it can capture invariants of human behavior and computation across problems and algorithmic-level implementations. In this thesis, I propose a way forward by introducing into the study of cognition, insights from computer science and in particular computational complexity theory (CCT)[6]. This is a branch of computing theory that studies the computational resource requirements of problems and the algorithms available to solve them.

The proposal of introducing CCT into the study of cognition is not new. Some researchers have advocated for the inclusion of the notion of computational complexity into models of human behavior and cognition (Frixione 2001; John K. Tsotsos 1990; van Rooij 2008; Bossaerts and Murawski 2017)[7]. The investigation documented in this manuscript builds on these proposals and introduces to cognitive science a closely related mathematical framework capable of quantifying hardness of tasks in order to generate empirically testable predictions. Importantly, I experimentally demonstrate the validity and capabilities of this framework for the study of human computation.

In the remainder of the introduction, I offer an overview of existing studies that investigate computational hardness from different vantage points. I first present a brief summary of current methods for quantifying difficulty of tasks related to human behavior. Afterwards, I introduce how metrics of computational hardness have been employed to discover generalities in the neural processes supporting problem-solving. Finally, I present an introduction to how hardness has been studied in CCT, focusing on the theoretical framework that will be employed here.

Subsequent chapters apply this theoretical framework to the study of cognition. In chapter 2, I define a set of metrics of computational hardness that will be considered, and propose a pipeline for the application of this theoretical framework to human cognition. In chapter 3, *we* apply this framework to the study of human behavior in one ubiquitous computational problem: the knapsack problem. In chapter 4, *we* demonstrate the generality of the proposed framework by experimentally testing and comparing the results across two additional canonical problems. Afterwards, in chapter 5, *we* apply this framework to the study of the neural processes associated with problem-solving. In the last chapter, I present some concluding remarks,

[6]In this manuscript I use the broad definition of *computational complexity* to refer to the study of computational resource requirements for solving a task (Arora and Barak 2009; Pudlák 2013; Moore and Mertens 2011; Moser, Gheorghita, and Aleti 2017). This notion is not to be confused with the definition of computational complexity in terms of classical complexity classes, in particular, complexity classes based on asymptotic worst-case analysis such as P and NP (see section 1.3.2).

[7]The reader might have, indeed, noticed that I alluded to this already when referring to the intractability of day-to-day tasks.

including implications of this work and possible future directions.

1.1 Computational hardness and behavior

Several lines of research have considered the effects of computational hardness on specific cognitive tasks. However, these studies characterize hardness in task-specific ways and, in many cases, specify hardness based on a particular cognitive strategy. This is problematic because of the variety of tasks that people face daily and the many strategies people might implement to perform them. In this section, I briefly review these approaches and highlight their limitations.

Computational hardness may vary not only from one task to another, but also from one instance of a particular task to another instance. For instance calculating whether a number is divisible by 3 may be hard if the number is large, but easy if it is small. Here, I focus on approaches that have been employed to characterize the hardness of instances of tasks.

A prominent approach for studying computational hardness is one that is built based on a particular strategy specification. From this perspective, hardness (or difficulty) is defined based on the computational steps (or operations) required by a particular strategy to solve a problem. This approach can potentially capture a significant amount of variance due to differences in strategy use. Consider, for example, the problem of determining whether the number $11,718$ is divisible by 3. The time (number of steps) required to solve the problem can vary depending on the strategy implemented. For example, the agent might be aware that it is possible to estimate divisibility by 3 by assessing the divisibility over the sum of the digits. Using this procedure, the divisibility of $11,718$ can be assessed by adding the digits ($1+1+7+1+8=18$) and then determining whether the sum (18) is divisible by 3. Alternatively, the agent might solve the problem by directly dividing $11,718$ by 3. Notably, these two methods would entail different computational requirements that could be estimated if the strategy implemented was known. Overall, if given a strategy and a problem, it would be possible to estimate the number of computational step needed to solve the problem.

This approach has been amply used to study computational hardness and its effects on human behavior (e.g., Murawski and Bossaerts 2016; Acuña and Parada 2010; Dry, Lee, et al. 2006; Guid and Bratko 2013; Fimbel, Lauzon, and Rainville 2009). This line of research categorizes hardness of a problem according to the number of computational steps or amount of time a particular algorithm needs to solve the problem. These hardness metrics can then be related to human behavior. Consider, for example, the traveling salesperson problem (TSP). In this problem, an agent is asked to visit a set of cities and return home while minimizing the travel distance (Fig 1.1a). Several studies have explored how strategy-specific metrics of hardness affect human behavior in this problem. It has been found that the performance of certain algorithms or heuristics (implemented on electronic computers) correlates with human performance (Dry, Lee, et al. 2006; MacGregor and Chu 2011; Hill 1982). Moreover, there is evidence to suggest that these metrics are related to how people explore the possible travel paths while solving the problem (Acuña and Parada 2010).

In this line of research, a particularly relevant problem that has been studied is the knapsack problem. This problem consists of filling a knapsack (backpack) that

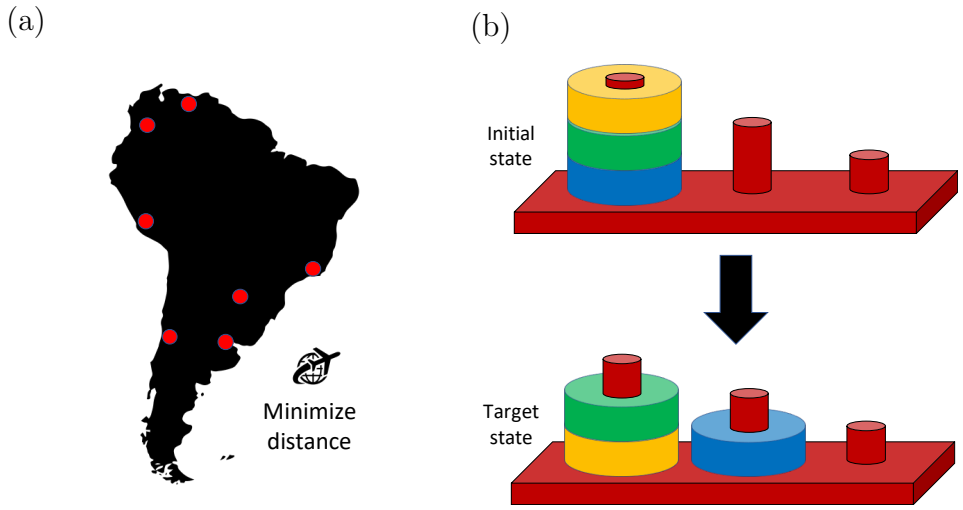


Figure 1.1: **Computational tasks. (a) The traveling salesperson task.** Given a set of cities in a map, the objective is to find a path to visit all the cities and return to the origin whilst minimizing the distance traveled. **(b) The tower of London task.** The objective is to find a sequence of moves to transform the initial state into the target state in the minimum number of moves. Each move corresponds to picking and dropping one disk from one rod to another.

has a specified weight capacity with predefined items that have a weight and value (Fig 1.2). The goal is to choose those items that maximize the total value in the backpack without exceeding the weight limit. Human performance in this problem has been linked to strategy-specific metrics of hardness. Specifically, it has been shown that the time people expend solving the problem and the quality of their solutions is affected by a metric of hardness called Sahni-k (Murawski and Bossaerts 2016), which is based on the Sahni-k algorithm (Sahni and Sartaj 1975).

Overall, this approach provides a way to explore computational hardness and its relation to human behavior at a granular level. This method can capture the variability in the number of computational steps (and memory requirements) needed to solve a problem depending on the strategy used. However, this approach ignores the diversity in algorithms that different people may use across contexts. Even if the computational problem is the same, different people might solve the problem using different procedures and might change their procedures depending on the situation or their level of experience with the problem (e.g., MacGregor and Chu 2011; Acuña and Parada 2010; Hirtle and Gärling 1992; Murawski and Bossaerts 2016; Ohlsson 2012; Gerd Gigerenzer and Gaissmaier 2011; Newell, Weston, and Shanks 2003; Payne, Bettman, and Johnson 1993). For instance, if the reader is now asked to find whether $111,111$ is divisible by 3, they might use a different strategy to the one they would have employed before reading this chapter. Comprehensive ways of quantifying the amount of computational resources needed to solve a problem are then particularly problematic given the lack of a generic strategy.

Alternative approaches have studied the computational hardness of a task independently of the strategy employed. These approaches presume the existence of intrinsic hardness of a problem. In other words, they assume that a task is inherently easy or hard and that this intrinsic feature has an effect on human performance independently of the strategy implemented.

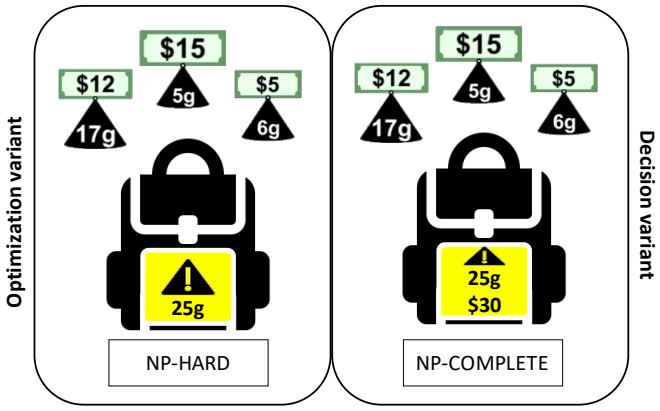


Figure 1.2: **0-1 Knapsack problem.** In the optimization variant of the problem (left), the objective of the problem is to select a subset of items, each with a weight and value, that maximizes the value packed into the knapsack (backpack) without exceeding the capacity constraint ($25g$). In the decision variant of the problem (right) the objective is to determine whether the target value ($30) can be packed without exceeding the capacity constraint ($25g$). The optimization problem belongs to the complexity class NP-hard and the decision variant belongs to the class NP-complete (these classes are defined in section 1.3).

A prominent example derives, as well, from the study of human behavior in the TSP. It has been shown that certain intrinsic features of the problem affect human performance (MacGregor and Chu 2011). Notably, the layout of the cities (for instance how clustered or regular they are) has been shown to affect human effort and solution quality (MacGregor, Ormerod, and Chronicle 1999; Hirtle and Gärling 1992; Dry, Preiss, and Wagemans 2012).

Another example of problem-specific metrics of hardness stems from research in planning and the Tower of London task (Fig 1.1b). In this canonical task, the participant is presented with a set of disks (balls) of different colors that are organized into different stacks (rods), the subject is then asked to pick and drop disks (one at a time) in order for the arrangement of the disks to match a target state. Participants are asked to do this in the minimum number of moves.

Difficulty in this task is generally characterized based on the intrinsic structure of the problem. Explicitly, hardness is quantified as the number of 'pick and drops' needed to reach the target state. That is, hardness is characterized based on the properties of the solution (i.e., the sequence that reaches the target state with minimum number of moves), an intrinsic feature of the computational problem underlying the task. Importantly, this metric has been related to human performance: the longer the optimal solution, the worse people perform (Berg et al. 2010). Moreover, this metric has also been successfully used to assess cognitive abilities across clinical populations and age (Shallice 1982; Fimbel, Lauzon, and Rainville 2009).

Other problem-specific metrics of hardness have been studied in a number of different tasks (Basso, Bisiacchi, et al. 2001; Chu and MacGregor 2011; Fedorenko, Duncan, and Kanwisher 2013; G. Gratton et al. 2018). These include tasks on insight such as the matchstick problems (Knoblich et al. 1999), as well as other problems like mental rotation (Shepard and Metzler 1971) and the vertex cover problem (Carruthers, Masson, and Stege 2012).

Problem-specific characterizations of hardness, however, lack generality, which

would hinder their applicability. It is not clear how to extend these metrics to other problems or if they are even related to underlying generic characteristics that makes particular cases of problems computationally hard across tasks. Consider, for example, the hardness metrics related to the TSP such as how clustered the cities are. They are generally based on the graphical representation of the problem. This specification of hardness is idiosyncratic to the problem and it is not clear how these metrics could be generalized to other problems. Moreover, this characterization of hardness could be specific to the representation of the problem. The tasks used to study behavior in the TSP present the problem graphically (Fig 1.1a), in line with the picturesque description of the problem via a traveling agent. However, the same problem can be represented by a matrix of distances between cities. If participants were to be presented with the matrix representation of the problem, it would by no means be obvious whether the corresponding matrices of non-clustered problems would entail better performance than instances with clustered cities. Overall, the specificity of metrics of hardness hinder their applicability given the plethora of problems people face everyday.

One generic metric of hardness in computational problems that has been investigated previously is problem size. Several studies have shown that human performance worsens as the size of the problem increases (e.g., Carruthers, Masson, and Stege 2012; MacGregor and Chu 2011; Dry, Lee, et al. 2006; van Opheusden and Ma 2019; Stazyk, Ashcraft, and Hamann 1982; De Visscher and Noël 2014). For instance, in the TSP, the number of cities has been shown to affect human performance (Hirtle and Gärling 1992; Dry, Lee, et al. 2006). Other examples where problem-size has been shown to affect human behavior include mental arithmetic problems (Stazyk, Ashcraft, and Hamann 1982; De Visscher and Noël 2014) and planning tasks (van Opheusden and Ma 2019). It is worth highlighting that the problem size can be characterized in several ways. For example, it can be estimated as the size of the instance of the problem, as it is usually done in the TSP (i.e., number of cities), or it can be characterized as the size of the state space (that is, the number of possible combinations or paths) (van Opheusden and Ma 2019; Murawski and Bossaerts 2016). The latter might provide a way of comparing hardness across problems since it captures a generic characteristic of the underlying structure of the search landscape (see chapter 6).

Size captures a generic dimension of computational hardness that has been shown to affect human behavior in several tasks. Nevertheless, there is significant variability in human performance for a fixed problem size (e.g., MacGregor and Chu 2011; Murawski and Bossaerts 2016), and thus, it is unlikely that this is the only *generic* source of computational hardness. This issue, however, remains unexplored: it is an open question whether there are additional generic dimensions of hardness that affect human performance. In this thesis I address this question.

1.2 Computational hardness and the brain

The previous section introduced current approaches studying difficulty (computational hardness) of cognitive tasks. Various metrics of difficulty based on those approaches have been shown to affect human decision quality and effort. However, quantification of hardness can also provide insights into the neurobiological basis of human problem-solving, that is, the characterization of the neural processes associ-

ated with problem-solving. Firstly, it provides a framework for studying the neural underpinnings of cognitive demand during problem-solving. Secondly, a generic quantification of computational hardness could shed light on the allocation of limited cognitive resources by presenting a metric that could be employed by an agent when making meta-decisions such as effort allocation.

At this point, relatively little is know about the neural processes involved in complex problem-solving. This is not to say that the neural processes of problem-solving are altogether unexplored. However, such work to date has focused mainly on easy problems.

Indeed, many lines of research have studied the neural processes associated with solving easy, or tractable[8], problems. Notably, problem-solving has been studied in relation to planning tasks (such as the Tower of London) (Ruocco et al. 2014; Nitschke et al. 2017), mathematics problems (Matejko and Ansari 2018), and insight problems (problems involving an "Aha!" moment) (Sprugnoli et al. 2017). Alternative lines of research have studied problem-solving in problems involving working memory updating (e.g., N-back task), task switching (e.g., Stroop task) and problems in which a prepotent tendency has to be withhold (e.g., inhibition tasks, go/no-go task) (G. Gratton et al. 2018; Miyake et al. 2000).

A prominent approach in the study of (tractable) problem-solving analyzes the neural processes involved, by studying the neural correlates of difficulty of a task (e.g., Fedorenko, Duncan, and Kanwisher 2013; Assem et al. 2020; Duncan and Owen 2000; Duncan 2010; Crittenden, D. J. Mitchell, and Duncan 2016). This approach allows for the study of the generic neural processes associated with problem-solving across different problems by implicitly defining a generic neural construct: *cognitive demand*. This construct can then be studied across tasks provided it reflects a real neural substrate related to the difficulty of a task. Importantly, this procedure minimizes confounding effects by avoiding the need to artificially produce alternative and arbitrary benchmark tasks. This framework has been particularly successful in characterizing a set of brain regions that generically correlate with difficulty across different tasks (Fedorenko, Duncan, and Kanwisher 2013; Assem et al. 2020; Duncan and Owen 2000; Crittenden, D. J. Mitchell, and Duncan 2016): the intraparietal sulcus (IPS), dorsal anterior cingulate cortex (dACC), the anterior insula (AI) and specific regions from the lateral prefrontal cortex including the inferior frontal sulcus and the middle frontal gyrus (MFG) (Duncan 2010; Fedorenko, Duncan, and Kanwisher 2013; Crittenden, D. J. Mitchell, and Duncan 2016). This collection of regions often takes the name of the multiple-demand system (MDS) (Fedorenko, Duncan, and Kanwisher 2013; Duncan 2010; Crittenden, D. J. Mitchell, and Duncan 2016).

The MDS overlaps with the cognitive control network. This is not surprising, given that their definitions are closely related. On one hand, MDS is characterized by modulating cognitive demand (i.e., difficulty) of a task. For instance, in working memory tasks, the cognitive demand of the task is generally modulated by varying the amount of information that needs to be maintained (e.g., Fedorenko, Duncan, and Kanwisher 2013; Assem et al. 2020; Duncan and Owen 2000). On the other hand, a prominent way to define the cognitive control network is by characterizing the neural correlates of 'cognitive control' in certain tasks. Prominent tasks used in this regard include: task switching tasks, conflict tasks, inhibition tasks and working

[8]See section 1.3 for a precise definition of tractability.

memory tasks (see G. Gratton et al. 2018; C. Gratton, Sun, and Steven E. Petersen 2018 for reviews). In working memory tasks (e.g., N-back task), cognitive load is manipulated by increasing the length of the sequence that needs to be maintained in working memory. In conflict paradigms, such as the Flanker task and the Stroop task, the level of cognitive load is manipulated by adjusting the level of conflict between task-relevant and task-irrelevant properties. Relatedly, in response inhibition paradigms, such as the go/no go task, a prepotent tendency has to be withhold. Finally, in the task switching task, the level of cognitive control needed is manipulated according to the frequency at which the objective task changes. Overall, this definition of cognitive control requirements matches the definition of difficulty used to characterize the MDS (Fedorenko, Duncan, and Kanwisher 2013; Camilleri et al. 2018), thus making both terms practically indistinguishable. Here I refer to these networks as the MDS, following Duncan 2010.

Among the set of flexible processes that support problem-solving in the MDS, two distinctive levels of processes are differentiated: *processors* and *controllers* (Jonathan D Power and Steven E Petersen 2013; Shenhav, M. M. Botvinick, and J. D. Cohen 2013; G. Gratton et al. 2018; C. Gratton, Sun, and Steven E. Petersen 2018; Posner and Steven E Petersen 1990; Miller and J. D. Cohen 2001). Processors, on the one hand, are units of computation that are specialized and do specific moment-to-moment operations. Controllers, on the other hand, orchestrate processors at a higher level such that they perform successfully the task at hand. Note that the distinction between control and processing operations is *per se* a nuanced one. Concretely, under these definitions, the MDS can be viewed as a network that supports generic problem-solving, which might encompass controllers and processors alike. *Some* of the MDS regions might be associated with implementing allocation of control (proactive control), while others might support basic processes. These basic operations might inform the proactive allocation of control (e.g., monitoring of counterfactual performance; Koechlin 2016) or might even be connected to other generic processes in problem-solving (e.g., memory retrieval).

Interestingly, the two theoretical processes supporting problem-solving mirror a network partition found in the MDS: the frontoparietal network (FPN) and the cingulo-opercular network (CON) (Crittenden, D. J. Mitchell, and Duncan 2016; Sadaghiani and D'Esposito 2015; Jonathan D Power and Steven E Petersen 2013; Marek and Dosenbach 2019; Dosenbach, Fair, Miezin, et al. 2007; Dosenbach, Fair, A. L. Cohen, et al. 2008; Nomura et al. 2010; Jonathan D. Power et al. 2011; Seeley et al. 2007). This separation between subnetworks in the MDS stems mainly from their functional connectivity profile. It has been identified that there is higher functional connectivity within than between subnetworks. Importantly, this differentiation has been found during resting state (Dosenbach, Fair, Miezin, et al. 2007; Dosenbach, Fair, A. L. Cohen, et al. 2008; Seeley et al. 2007; Jonathan D. Power et al. 2011) as well as in task-related connectivity during problem-solving (Crittenden, D. J. Mitchell, and Duncan 2016).

Additionally, there is evidence to suggest that the two networks have different characteristic temporal response profiles during problem-solving and that they are associated with dissociable processes. FPN, on one hand, has been associated mainly with trial-by-trial transient responses that are believed to encode error related activity as well as control instantiation (Marek and Dosenbach 2019; Dosenbach, Fair, Miezin, et al. 2007; Dosenbach, Visscher, et al. 2006; Dosenbach, Fair, A. L. Cohen,

et al. 2008; Jonathan D Power and Steven E Petersen 2013). On the other hand, it has been suggested that CON is predominantly involved in the sustained task-set maintenance across successive trials in a task, as well as transient signals related to task-set selection and instantiation (Marek and Dosenbach 2019; Dosenbach, Fair, Miezin, et al. 2007; Dosenbach, Visscher, et al. 2006; Dosenbach, Fair, A. L. Cohen, et al. 2008; Sestieri et al. 2014; Jonathan D Power and Steven E Petersen 2013). CON has also been shown to encode a set of trial-specific signals related to the level of conflict (M. Botvinick et al. 1999), errors (Neta, Nelson, and Steven E Petersen 2017; Neta, Schlaggar, and Steven E Petersen 2014; Dosenbach, Visscher, et al. 2006), ambiguity (Bossaerts 2018; Neta, Nelson, and Steven E Petersen 2017; Neta, Schlaggar, and Steven E Petersen 2014), tonic alertness (Sadaghiani and D'Esposito 2015; Coste and Kleinschmidt 2016) among others.

Overall, the main distinction between sub-networks is grounded in FPN not encoding sustained task-set maintenance signals (Jonathan D Power and Steven E Petersen 2013; Marek and Dosenbach 2019). Critically, evidence suggests that CON encodes both sustained signals as well as the transient signals found in FPN (Jonathan D Power and Steven E Petersen 2013; Dosenbach, Visscher, et al. 2006). This characteristic property of CON has led it to be denominated the 'core' network of control (Dosenbach, Visscher, et al. 2006).

This core network would then be involved in the endogenous process of recruiting task-specific processors conditional on the task at hand. A prominent view in the investigation of this endogenous process asserts that the dACC (which is part of CON) plays a leading role in the orchestration of this process (Shenhav, M. M. Botvinick, and J. D. Cohen 2013; Dosenbach, Visscher, et al. 2006; Silvetti et al. 2018; Vassena, Holroyd, and Alexander 2017; Holroyd and Yeung 2012; Alexander and Brown 2011). Evidence in this regard has shown an increase in functional connectivity between dACC and task-relevant areas during problem-solving. This pattern has been shown specifically in perceptual (Sestieri et al. 2014; Aben et al. 2020; Crottaz-Herbette and Menon 2006) and memory tasks (Sestieri et al. 2014).

While CON seems to be predominantly involved in generic control processes, FPN involvement in problem-solving might be associated with a more nuanced interplay between generic controllers and processors. In particular, it has been suggested that the involvement of FPN in control signals is due to its overlap with regions associated with task-specific 'processing' units. For instance, IPS, which is part of FPN, has been strongly associated with processing of numerical magnitudes (Matejko and Ansari 2018; Brannon 2006; Arsalidou and M. J. Taylor 2011) and FPN, more comprehensively, has been linked to processing in mathematical calculations (Matejko and Ansari 2018; Grabner et al. 2009; De Smedt, Holloway, and Ansari 2011; Arsalidou and M. J. Taylor 2011).

Despite the extensive research on problem-solving and its neural underpinnings, it remains an open question as to what are the neural processes that support complex problem-solving. This is particularly troublesome because there is no obvious way to define *cognitive demand* in complex problems. To date, the only studies that have explored this set of problems have employed the map task (Basso and Saracini 2020; Basso, Lotze, et al. 2006; Basso, Bisiacchi, et al. 2001). This task presents a specific version of the TSP on a grid, which has been suggested to be intractable (i.e., NP-hard)[9]. However, previous studies using this task have two main

[9]This task, denominated the *maps task*, corresponds to a restricted version of the TSP in which

limitations. Firstly, previous work has lacked spatial resolution since these studies have explored the neural underpinnings via a lesion study and transcranial magnetic stimulation. Secondly, and more importantly, the approach used to study this tasks utilizes features of the task that are highly problem-specific (Basso, Bisiacchi, et al. 2001) or that are even based on the assumption that a specific set of predetermined strategies are being used (Basso and Saracini 2020; Basso, Lotze, et al. 2006). This approach, as mentioned in the previous section, is difficult to generalize given the diversity of problems and strategies to consider. Moreover, the specification of task-specific strategies makes generalization across tasks difficult. Even extending these results to different presentations of the TSP (e.g., non-grid TSP) is not straightforward. What would be particularly desirable is a generic framework to characterize hardness, and thus cognitive demand, in complex problems. The framework put forward in this thesis allows for such characterization. For instance, it allows for the study of the TSP without the need to limit the task to a grid and without the need of assumptions about the strategies employed by agents (see chapter 4 for an application of this framework to the TSP).

Besides the complications that arise when quantifying cognitive demand of complex problems, the exploration of these problems entail another inherent complication. Complex problems usually require more time to solve, and thus, the neural processes that support problem-solving can no longer be modeled as a static system. The successful characterization of the neural underpinnings of complex problem-solving needs to take into account that this process ensues from a dynamic interplay of neural activity that generates strategies, modulates cognitive effort, all whilst keeping track of relevant markers of performance such as expected error (Neta, Nelson, and Steven E Petersen 2017; Bossaerts 2018), expected rewards (Duverne and Koechlin 2017), the level of uncertainty (Neta, Nelson, and Steven E Petersen 2017; Neta, Schlaggar, and Steven E Petersen 2014; Bossaerts 2018), among many other possible markers (Yoo, Hayden, and Pearson 2021; Koechlin 2016). To date, the neural invariants of this dynamic interplay during complex problem-solving have not been investigated.

Although there is robust evidence to suggest that there is generic (task-independent) involvement of FPN and CON in problem-solving (Fedorenko, Duncan, and Kanwisher 2013; Assem et al. 2020; Duncan and Owen 2000; Duncan 2010; Crittenden, D. J. Mitchell, and Duncan 2016; G. Gratton et al. 2018), the precise response profile might be contingent on the task at hand. For instance, Dubis et al. 2016 presented results that suggest that the sustained activation of CON is not present in perceptual tasks unless the task requires further cognitive processing beyond stimulus perception. This suggests that some of the signals associated not only with FPN, but also with CON, are task-specific. Notably, it remains an open question whether CON is involved in proactive control during complex problem-solving.

In summary, extensive research has studied the neural processes that support problem-solving. This program has characterized a network of regions that respond robustly to cognitive demand regardless of the task at hand: the MDS (Duncan 2010; Fedorenko, Duncan, and Kanwisher 2013). This has been done using a number of tasks including perceptual target detection and memory retrieval, among

the locations to visit are located on a grid. Technically, the problem is a TSP on solid grid graph with Manhattan distance. This problem has been conjectured to be NP-hard, but there is currently no proof of it(Demaine, J. Mitchell, and O'Rouke 2004).

many others. Notably, most of the tasks employed to date are based on tractable problems. Moreover, many of the tasks employed modulate cognitive demand of the task by tuning the amount of processing needed in one specific dimension of cognitive processing, for instance, by manipulating working memory demands or signal strength in perceptual tasks (G. Gratton et al. 2018; Fedorenko, Duncan, and Kanwisher 2013; Crittenden and Duncan 2014).

The lack of a generic (task-independent) definition of cognitive demand prevents the generalization of this approach to new problems. Importantly, this approach, as it stands, is restricted to the ordinal study of cognitive demand modulation within a task. Moreover, the level of cognitive demand might be highly related to the strategies used. For instance, multiplication operations can be performed using different strategies such as the standard multiplication algorithm or alternatives such as the Japanese visual method and the Vedic method (Garain and Kumar 2018). These different strategies would predict different levels of cognitive demand in multiplication problems depending on the algorithm used. Critically, when leaping into tasks that are more complex and especially those that involve problems that are intractable, the limitations of this approach become more apparent. The increase in complexity brings along an increase in strategies available to solve complex problems and this renders the determination of a single metric of cognitive demand even more troublesome.

A proper quantitative (cardinal) study of the neural underpinnings of cognitive demand requires a proper generic definition of cognitive demand that can be quantified across problems and, ideally, across strategies. Such characterization would be grounded in the assumption that hardness is, at least partially, an intrinsic characteristic of the problem at hand. In this thesis, I take this approach and present a framework able to quantify intrinsic computational hardness of tasks in a generic way. I conjecture that with this framework it is possible to characterize neural invariants of complex problem-solving, that is, capture cognitive demand and neural markers related to the structure of the task.

The generic framework put forward is particularly valuable in the study of allocation of control during problem-solving given the generality of this process. The human brain has limited cognitive resources, yet is able to reuse and reallocate resources in order to successfully solve a plethora of problems. A true understanding of the human brain and its neural processes would require a generic model capable of generalizing across the specifics of a task. In this line, several proposals have been put forward in which the allocation of limited cognitive resources is modeled as a mechanism in which a generic construct of cognitive demand is estimated and effort (or control) is consequently allocated based on this characterization (e.g., Shenhav, M. M. Botvinick, and J. D. Cohen 2013; Verguts, Vassena, and Silvetti 2015; Westbrook and Braver 2015). Such a construct would presumably need to be characterized from generic features of the task such that the agent is able to estimate cognitive demand across tasks. In this thesis, I define a potential component of this construct: computational demand.

Overall, current approaches in neuroscience and cognitive sciences have considered the computational hardness of tasks in ways that are task- or strategy-specific. To date, there is is no theoretical framework to study computational hardness in cognition in a generic way. Computational complexity theory might provide such a framework. This field encompasses several approaches that have been extensively

used to study the hardness of problems in computer science. Importantly, it allows for the study of hardness across different problems without the need to assume a particular procedural strategy used to solve a problem. In the next section I provide a select overview of this theory.

1.3 Theory: Quantifying intrinsic computational hardness

Computational complexity theory (CCT) studies the computational resource requirements of problems and the algorithms available to solve them. In other words, it studies how efficiently problems can be solved by computers. This mathematical framework provides a principled way of studying computational hardness of cognitive tasks by characterizing tasks as problems and agents as computers. Characterization of cognitive tasks as computational problems allow us to build on computing theory to operationalize computational hardness. Critically, this theory provides a mathematical framework to examine cognition in a principled way. Under this characterization, cognitive tasks are abstracted into a mathematical form that capture the current state, the structure of the task at hand and the objective state. Moreover, the cognitive process through which the task is performed can also be explicitly defined using this framework with the notion of computation. Overall, this theory introduces specific terminology, definitions and results that avoid impreciseness and can elucidate research on cognitive processes and the resulting behavior. In this chapter, I define the terminology utilized throughout this manuscript and explicitly state the underlying hypotheses behind the application of this framework to human cognition. Afterwards, I give a brief overview of the CCT framework from the most prominent approach to that which I will then employ to quantify computational hardness.

1.3.1 Computing theory for the study human cognition?

The question that inevitably arises is whether the theory of computation can be used as a framework to study human cognition. This would require assumptions on how computation relates to cognition. Indeed, in this manuscript I take a computationalist approach (Rescorla 2020) to studying cognition and decision-making. I start from the premise that humans are in fact computers.

When people think of computers, many might think of a laptop or even a smartphone. However, I am not suggesting that humans are electronic computers, but rather that cognition can be studied through theoretical notions of computation. More specifically, I adhere to a tradition of cognitive scientists that considers that it is possible to analyze cognitive processes and behavior by characterizing the human agent as a computer with limited computational capacities (van Rooij, Blokpoel, et al. 2019; Bossaerts and Murawski 2017; John K Tsotsos 1988; Frixione 2001; Blum and Vempala 2020; Aaronson 2013). Here, instead of addressing this philosophical question at a conceptual level, I take an empirical perspective and test experimentally the value of this approach in capturing invariants of cognition and decision-making. However, in order to explain what are the precise assumptions at a theoretical level, I will first aim to introduce some key concepts.

Theory of computation

The theoretical framework, and corresponding definitions, implied can be illustrated by considering a calculator. In this computer, an input such as "4×6" will generate an output "24". The input in this example corresponds to the three symbols in their order ("4","$\times$" and "6"), which effectively encodes the structural definition of the *problem* (multiplication) and the particular *instance* of the problem to be solved (operands "4" and "6"). After receiving the input, the calculator implements a set of basic steps that are predetermined by a list of instructions. This list of instructions is called an *algorithm*.

The formal definition of a problem is then that of a mapping $f : I \mapsto O$, and an instance is a particular input to that mapping: $i \in I$. The effective procedure through which i is transformed to $f(i)$ is described by the notion of computation. The operationalization of this notion encapsulates the set of permissible step-by-step operations that represent an algorithm. A ubiquitous mathematical model of computation is captured by the Turing machine. This mathematical model of computation has been proposed to be an all-inclusive representation of computation. The latter claim is known as the Church-Turing thesis: the conjecture that *any* input-output mapping that can be computed can be also computed by a Turing machine (Church 1936; Turing 1937). If the conjecture were to be true, it would hold for *any* type of computer, including quantum computers, analog computers, reservoir computers, and most importantly the brain. Although it is a conjecture, the Church-Turing thesis is widely accepted as true (Pudlák 2013; Arora and Barak 2009; van Rooij, Blokpoel, et al. 2019).

Note, however, that the implicit hypothesis put forward in this manuscript is not this conjecture, but a related one. The Church-Turing conjecture speaks to the feasibility of computing (i.e., computability) a particular mapping (i.e., problem). It considers the ability of any computing system on doing an input-output operation assuming unlimited resources (e.g., unlimited memory, unlimited time). As such, it is silent with regards to the implementation of the algorithm, despite the fact that the Turing machines are closely related to one specific algorithm-implementation mechanism: the von Neumann architecture. Being silent about the particular implementation mechanism makes it, arguably, silent about the resource requirements of implementing that algorithm. Since the object of analysis in this thesis is the effect of resource requirements of tasks on human computation, the hypothesis put forward here differs from the Church-Turing conjecture. Specifically, I hypothesize that the cognitive procedures by which humans perform these mappings are affected by the structure of the problem (f) and the instance (i), just like the procedures implemented by Turing machines and related computing devices are affected by the structure of the problem at hand[10]. In a way this presumes the existence of computational hardness that is driven by intrinsic properties of the problem and their instances, and which affect computing procedures across computational models.

Since computational hardness of problems (and instances) are ultimately as-

[10]It is worth highlighting that this hypothesis differs from the so-called invariance thesis (Frixione 2001), which conjectures that any two reasonable (i.e., realistic) computational implementation mechanisms vary in time-requirements in at-most polynomial-time. The invariance thesis is an asymptotic notion which makes predictions about the problem complexity (see section 1.3.2) associated with different computing machines, but does not make granular predictions about the relative computational requirements of *instances* of a problem.

sessed based on the computational resource requirements of computing procedures, it then becomes critical to understand the alternative procedures to which cognitive procedures will be compared. These alternative procedures are encapsulated in the notion of algorithms. I turn now to describing different characterizations of algorithms. This will allow me to avoid vagueness when describing characteristics of cognitive procedures by referring to their algorithmic counterpart, which have explicit mathematical definitions.

There are many types of algorithms. To illustrate this, consider again the previous example. If the calculator always generates the correct solution ("24") then the algorithm is called an *exact algorithm*. If the algorithm does not guarantee the solution, but does guarantee to not deviate *'too much'* from the solution, then it is called an *approximation algorithm*. Note that the approximation guarantee can be defined in many ways, but the overarching definition still holds. For instance, an approximation dimension can be defined based on a maximum distance (ϵ) between the output ($f(i) = o$) and the correct solution (24); that is, $|24 - o| < \epsilon$. Another reasonable addendum to approximation algorithms are those that have an approximation guarantee that is stochastic. Specifically, these types of approximation algorithms would guarantee a minimum likelihood that the algorithm reaches the correct solution (e.g., $P(o \neq 24) < \epsilon$). A related notion, which is worth highlighting due to its prevalence in models of cognition are those algorithms that do not have an approximation guarantee. These are commonly referred to as *heuristics*. Is is important to note that the term heuristics in cognition usually connotes that the output will not deviate *'too much'* from the solution, but without any formal definition of the approximation guarantee[11].

It is worth highlighting that both the algorithm type and the problem specification are intertwined. Specifically, we can incorporate the distinction between exact and approximation algorithms directly into the algorithm or into the problem. For instance, in the calculator example a new problem specification, call it *ϵ-multiplication*, can be generated based on standard multiplication such that the solution of *ϵ-multiplication* is defined as any number o that is at most ϵ-away from the standard multiplication solution. In this case every approximation algorithm for the *standard multiplication* problem could be described as an exact algorithm for the *ϵ-multiplication* problem. This entails that the approximation characteristic can be ascribed to the problem or the algorithm in an analogous fashion. Here I will focus my attention on exact algorithms, but this equivalence will allow me to make generalizations at a later stage (see section 6.1.2).

1.3.2 Computational-complexity theory

CCT studies the amount of computational resources needed to solve a problem or an instance of the problem. The amount of computational resource requirements characterizes the *computational complexity* (*computational hardness*)[12]. This can be done with or without reference to a particular algorithm. On one hand, CCT studies the intrinsic computational requirements implied by a problem independently of the algorithm used. On the other hand, this theory explores the computational re-

[11] See van Rooij, Wright, et al. 2018 and van Rooij, Blokpoel, et al. 2019, (chapters. 8 and 9) for a discussion on this.

[12] I will refer to these two terms interchangeably.

sources needed by a particular algorithm to solve a specific problem. These different approaches to study computational hardness represent different levels of analysis.

Computational hardness can, thus, be studied at the level of the agent (computer) or at the level of the structural properties of the task (problem or instance). I take the latter approach. This is not to say that the the study of the computational requirements of an algorithm is not relevant in the investigation of human cognition, but rather represents a specific decision made on the object of analysis [13]. Explicitly, I aim at characterizing invariants of human computation (Simon 1990) that relate directly to the intrinsic computational hardness of a task. Critically, the existence of such invariants is an empirically testable hypothesis that, if true, would imply the existence of intrinsic hardness of tasks that have effects across different models of computation (Aaronson 2005; Yadav et al. 2020; Blakey 2011).

In this section, I present a set of alternative approaches to those introduced in previous sections. These capture intrinsic hardness of tasks in a generic (task-independent) fashion. Among the approaches in CCT that study the intrinsic computational hardness of tasks in a generic way, the most prominent one investigates the computational requirements of problems. In the following sections I start with a description of this canonical approach and then move to alternative approaches that have studied the computational hardness of instances of problems, which are the foundations of the framework that is employed here.

Problem complexity

The main branch of CCT has studied the computational hardness at the level of problems (e.g., multiplication). Here, computational requirements are analyzed from an asymptotic perspective, that is, how fast resource requirements increase as the size of the input of the problem increases. This means that resource requirements are characterized in terms of their growth as a function of the input size of the problem (e.g., the number of digits of the operands in the multiplication problem). Problems with similar resource requirements, thus defined, are then grouped into complexity classes (Arora and Barak 2009).

The most widely used approach from this asymptotic perspective, categorizes problems according to the computational time required to solve the most difficult problem given an input size (Pudlák 2013; Moore and Mertens 2011; Arora and Barak 2009). Other approaches include average-case complexity analysis (Bogdanov and Trevisan 2006) and space complexity analysis, which studies memory requirements instead of time (Arora and Barak 2009). Note that time and space here are defined based on the notion of computation characterized by the Turing machine. Time is then defined as the number of operations performed by the Turing machine, while memory is defined as the amount of memory space (i.e., length of tape) required by the Turing machine to solve the problem. Overall, these specific characterizations of computational requirements imply an assumption about the computational model, which is generally the Turing machine.

In one of these approaches lies the theoretical framework from which the more common complexity classes arise (Fig 1.3). Explicitly, the *asymptotic worst-case time-complexity* is the framework employed to define complexity classes such as P and NP, and in general, to refer to the tractability of problems. Class P is defined

[13] See section 1.1.

as the set of problems that can be solved by a Turing machine in polynomial time. In other words, given an input of size n, the number of computational steps needed by a Turing machine to guarantee reaching the solution scales at a lower rate than a polynomial function of n. Similarly, class NP is defined as those problems for which an algorithm exists that could be solved in polynomial time by a non-deterministic Turing machine. Non-deterministic Turing machines are defined as Turing machines that can be in multiple states at the same time. Put simply, when this machine is searching for a solution it can simultaneously explore multiple search paths and return a solution whenever *any* of the paths finds the solution[14]. In contrast, a standard (deterministic) Turing-machine can only search one path at a time. Given these definitions, it follows that $P \subseteq NP$; moreover, it is conjectured that $P \neq NP$, yet no proof of this exist to date (Arora and Barak 2009; Pudlák 2013; Moore and Mertens 2011).

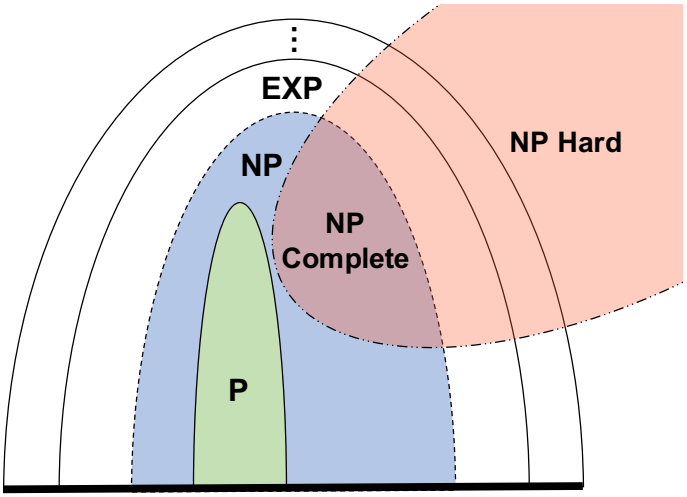


Figure 1.3: **Computational problem-complexity classes.** The classes depict the asymptotic worst-case time-complexity classes. Tractable problems are generally considered to be in class P, while NP-hard problems are deemed to be intractable. The dashed lines indicate the open question of P vs. NP. If $P = NP$, then all NP and NP-complete problems would collapse into P.

The notion of tractability is usually based on these complexity classes. Specifically, *tractable problems* are commonly defined as those problems belonging to the class P. In contrast, the class of problems that are considered to be *intractable* are those belonging to the class *NP-hard*. This class includes all those problems that are *at least as* difficult as all problems in NP. Cook 1971 and Levin 1973 were the first to define this class; explicitly, they proved that the Boolean satisfiability problem (SAT) is NP-hard. From that result, any other NP-hard problem can be proven to belong to this class by showing that it is harder than SAT; or more specifically, that it can be reduced in polynomial time to SAT. Put simply, this entails that the problem under consideration can be solved by solving a corresponding instance of the SAT with the help of a transformation algorithm, which is in P. An additional sub-class of NP-hard problems which will be widely alluded to in this manuscript

[14]These machines are entirely hypothetical.

is the class NP-complete. This class is composed of those problems that belong to both class NP and class NP-hard.

Many problems have been shown to belong to class NP-hard, including the 0-1 knapsack optimization problem (Kellerer, Pferschy, and Pisinger 2004) and the traveling salesperson problem (Arora and Barak 2009). Importantly, these and several other optimization problems of class NP-hard have a corresponding problem in class NP-complete. Consider for instance the 0-1 knapsack optimization problem. This problem consists of filling a knapsack (backpack) that has a specified weight capacity with predefined items that have a weight and value (Fig 1.2). The goal is to choose those items that maximize the total value in the backpack without exceeding the weight limit. A corresponding NP-complete problem is the knapsack decision problem. In the structure of the problem is the same as before, except that the objective is not to find the maximum value that can be packed, but to determine whether a specific target profit can be packed without exceeding the capacity (Fig 1.2). While the first problem is an optimization problem, the latter is a *decision problem*, that is, a problem whose answer is either 'yes' or 'no'. Note that the knapsack optimization problem would be considered harder than the decision problem because solving the optimization variant effectively solves the decision variant (with any target profit).

It has been proposed that problem-complexity classes can be used to shed light on models of cognition. Specifically, it has been proposed that models of cognition ought to be tractable. According to this conjecture, human cognitive capacities are insufficient to solve intractable problems reliably (Frixione 2001; John K. Tsotsos 1990; John K Tsotsos 1988). However, this conjecture, which was later coined the P-Cognition thesis, is considered too restrictive (van Rooij 2008).

A refinement of this framework that addresses this issue is known as the FPT-cognition thesis (van Rooij 2008; van Rooij, Blokpoel, et al. 2019). This thesis states that the notion of tractability in the classical sense (i.e., P class) is too limiting and that an alternative notion should be considered, namely, fixed parameter tractability. This approach proposes that in many real-life scenarios, the problems faced might be small in the size of certain parameters of the problem. In turn, these parameters can potentially be the source of hardness of the problem and thus, when constraining the size of these parameters of the problem, the problem becomes tractable. In other words, the intractability is constrained to this specific parameter, and thus can potentially be computed by a human agent when this parameter is small.

Both the P-cognition and the FPT-cognition proposal allow for the theoretical study of feasibility of computational-level models of cognition based on their computational hardness. Specifically, they specify for which models there could exist strategies (i.e., effective procedures) that always solve the problem within a reasonable time. This program, however, can not be used to explain differences in performance on different instances of the same problem. It categorizes problems based on the computational resources needed to solve the most difficult instance of the problem and according to the resources' asymptotic behavior. That is, problems are categorized based on how fast the computational resources increase as the size of the problem increases to infinity. They study computational requirements of problems based on their *asymptotic worst-case time complexity*. Admittedly, the FPT approach can be used to find features that are sources of worst-case asymptotic complexity of a problem. However, this approach does not consider if and how the features of particular instances of a problem affect human behavior and computation

on a single instance.

In order to make empirically testable predictions of human behavior based on metrics of intrinsic hardness, one must consider the computational hardness of individual instances. Consider for example the divisibility-by-3 *problem*. If a problem-complexity approach were to be used, all the instances of the problem would have the same complexity. Determining whether 9 is divisible by 3 and whether 1733 is divisible by 3 would have the same computational hardness. This raises the issue of whether there are more finely grained metrics of complexity that can be used to study human decision-making and cognition in general.

This section attempted to provide a brief summary of the literature on problem complexity. This framework encompasses the well-known approach based on which notions of tractability (P) and intractability (NP-hard) are defined. This is done, specifically, by categorizing problems into classes according to their asymptotic worst-case time complexity, employing a notion of computation encapsulated by the Turing machine. This approach can be applied to assess models of cognition, but is not amenable to the study of human behavior directly. Critically, this approach is limited by its lack of granularity. That is, it can categorize problems into classes of complexity, but it is unable to characterize differences in hardness across instances of these problems. Moreover, it characterizes hardness based on the growth rate (e.g., polynomial) as a function of the input-size. It is silent about the *level* of hardness of instances of problems with small size. Therefore, this theory is not suitable for the aim of this manuscript: characterizing empirically testable predictions from metrics of intrinsic hardness of individual instances. An alternative theoretical framework has studied hardness of instances of problems by using insights from statistical physics. I introduce this theory in the following section.

Typical-case complexity

A prominent framework in computer science investigates the drivers of computational hardness in computational problems by studying the difficulty of randomly generated instances of those problems. This line of research has revealed that there is substantial variance in the computational resource requirements for solving instances with the same input length. Importantly, this variability in hardness has been related to various structural properties of instances (Remi Monasson et al. 1999; Cheeseman, Kanefsky, and W. M. Taylor 1991; Ian P. Gent et al. 1996). The link between structural properties of an instance and the expected computationally complexity is commonly referred to as typical-case complexity (TCC). In this section I present an overview of this literature.

Definition of TCC

TCC maps a set of features of an instance of a problem to the expected computational requirements:

$$TCC : \alpha \mapsto \text{Computational Requirements}$$

In order to introduce this mapping, I need to describe both the properties of the instances that predict computational requirements (hardness) as well as the approaches used to estimate these requirements. First, I describe the canonical structural properties of the task that have been considered for this mapping and,

afterwards, I introduce how computational requirements are characterized in this framework.

Structural properties: Constrainedness and thresholds

The framework has been employed to characterize the computational hardness of *constraint satisfaction problems*. These problems involve a mathematical question in which a set of objects, which can take one of several states, is presented together with a set of constraints. The objective in these problems is to determine whether there exists at least one state in which all of the constraints are satisfied. Examples of these problems include the knapsack decision problem, the Boolean satisfiability problem and the traveling salesperson decision problem. Note that constraint satisfaction problems are a type of *decision problem*, and thus, their answer is either 'yes' or 'no'.

This research has characterized a structural parameter that is closely related to computational requirements. Specifically, it has been found that a so-called *constrainedness* parameter (α)[15] predicts the expected computational requirements of a typical instance of the problem. In other words, this parameter captures the average hardness of a *random ensemble* (i.e., a random sample) of instances. Note that this involves defining a specific random instance generation process, and thus, the parameter is determined not only by the instance properties, but also by the instance generation process.

The constrainedness parameter (α) captures how lenient or restricted the constraints are for a fixed input size n. The parameter determines the *satisfiability probability* (that is, the probability that the solution of the instance is *yes*), as well as the expected number of *solution witnesses*, that is, the expected number of variable configurations (e.g., combinations, paths) that satisfy the constraints (Ian P. Gent et al. 1996; Zweig, Palla, and Vicsek 2010; Rémi Monasson and Zecchina 1996; Ian P Gent and Walsh 1996; Cheeseman, Kanefsky, and W. M. Taylor 1991). Consider for example the knapsack decision problem. The constrainedness parameter can be characterized based on how restricting the target profit constraint is. If the target profit (α^p) is high, the instance is *overconstrained* and the chances of there being a combination of items that satisfy the constraints will be low. If, on the contrary, the target profit is low, the instance is *underconstrained* and the chances of finding a witness will be high. Indeed, in an underconstrained instance there might be many solution witnesses, that is, many combinations of items that satisfy the constraints (Fig 1.4).

Recent studies have found a plethora of asymptotic thresholds on the α parameter that are related to the hardness of an instance. Importantly, a phase transition has been identified in the satisfiability probability of many problems (Remi Monasson et al. 1999; Ian P Gent and Walsh 1996; Achlioptas, Naor, and Peres 2005; Yadav et al. 2020). In other words, this probability exhibits a discontinuity at a threshold (α^{sat}) at which the probability jumps from 0 to 1. The constrainedness value at which this phase transition occurs has been related to an increase (on average) in the computational requirements of solving an instance (Cheeseman, Kanefsky, and W. M. Taylor 1991; Ian P Gent and Walsh 1996; Selman and Kirkpatrick 1996;

[15]This parameter is usually referred to as the *order parameter*. I diverge from this terminology in order to highlight the specific dimension of hardness that I am referring to, that is, the level of constrainedness of the problem. I borrow this terminology from (Ian P. Gent et al. 1996).

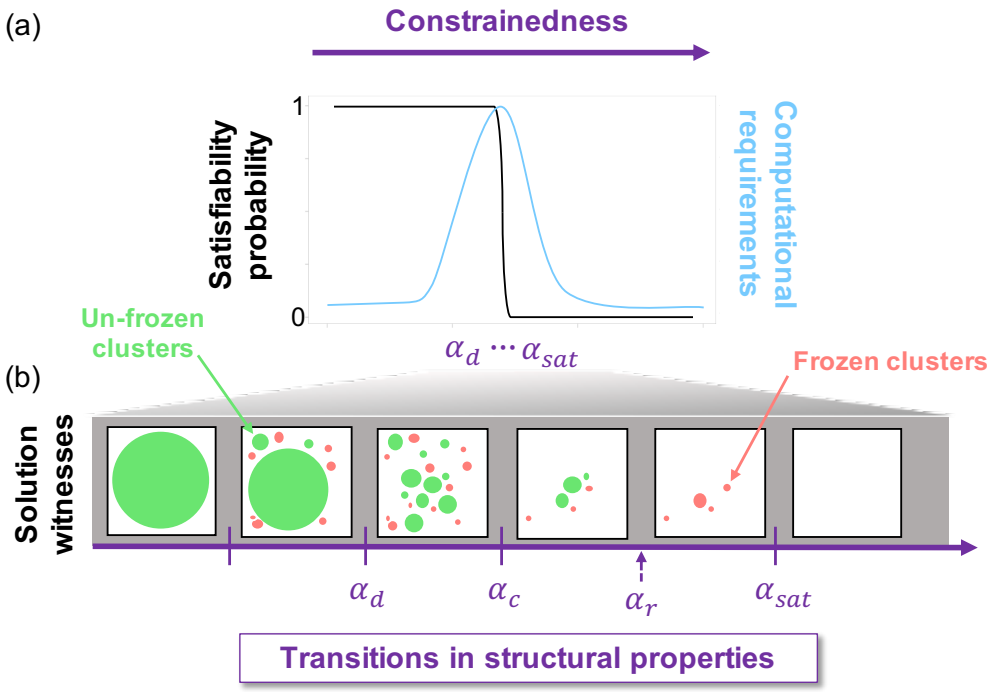


Figure 1.4: **Constrainedness, phase transitions and computational requirements in the thermodynamic limit.** Illustration of the transitions in the structural properties of an instance as the level of constrainedness (α) increases. (a) The satisfiability probability shows a jump from 1 to 0 in a narrow range of the α parameter. Around this narrow range of values of α the expected computational requirements of solving the instance is highest. However, there are several other values of α that have been related to increased computational requirements. These thresholds represent changes in the structural properties of the solution space. (b) At low values of constrainedness, witnesses are in a single cluster. This cluster starts gradually splitting until the dynamic threshold (α_d) is reached. Here, the solution space breaks into an exponential number of clusters. The number of clusters then collapses to a few clusters at the condensation threshold (α_c). Then, after the rigidity threshold α_r most of the witnesses belong to clusters with frozen variables. The clusters containing frozen variables are peach-colored and those that do not contain frozen variables are green. Finally, at the satisfiability threshold $\alpha_s at$ witnesses cease to exist, and thus, instances are unsatisfiable. It is worth noting that the order of these thresholds might differ across problems. Markedly, α_r might occur before or after α_c.

Yadav et al. 2020) (Fig 1.4a).

Note that the *satisfiability threshold* (α^{sat}) is defined in the limit as the location of the phase transition as $n \to \infty$. However, the exact definition of this phase transition is not agreed upon. There is evidence to suggest that the steep jump in the probability curve does in fact represent a phase transition in the structure of the instance (Rémi Monasson and Zecchina 1996). Indeed, different problems have been shown to have different levels of curve steepness, and the rate at which the steepness increases as $n \to \infty$ has been found to have an effect on the complexity class of the problem (Remi Monasson et al. 1999). However, there is also evidence to suggest that the so-called phase transition in the satisfiability probability represents instead

a sharp threshold related to the law of large numbers (Zweig, Palla, and Vicsek 2010). Irregardless of the terminology and underlying structural changes, the fact remains that instances around this threshold are on average harder to solve.

Besides the satisfiability threshold, several other thresholds have been associated with increased computational requirements. These, in general, represent locations of α at which a measure μ of a particular property of the instance undergoes a phase transition in the thermodynamic limit ($n \to \infty$ with α held fixed). The satisfiability probability is one such measure; however, many others have been studied. Many of them capture asymptotic properties of the solution space of random ensembles. That is, they capture topological properties of the set of solution witnesses of an instance (Fig 1.4b). This set is generally given a topological structure by defining nodes as witnesses and edges that connect the witnesses that differ in only one variable assignment. For instance, in the knapsack problem a witness is a packing combination of items that satisfy the constraints. In this example, two solution witnesses are connected by an edge if a single operation of including or excluding an item from the knapsack transforms one witness into the other. From this definitions several thresholds have been identified in many problems. For example, a dynamic (clustering) threshold α_d has been characterized where the solution space splits into multiple clusters (Krzakala, Montanari, et al. 2006; Krzakala and Zdeborová 2007). Another relevant threshold identified is the condensation threshold α_c, which marks a shift in the number of number of clusters in the solutions space (Krzakala, Montanari, et al. 2006; Krzakala and Zdeborová 2007; Zdeborová and Mézard 2008). Other thresholds include the freezing (α_f) and the rigidity (α_r) threshold (Krzakala and Zdeborová 2007; Semerjian 2008; Zdeborová and Krzakala 2007; Ardelius and Zdeborová 2008).

Hardness: Computational requirements

So far, I have focused on how the constrainedness parameter (α) is defined and the corresponding thresholds that determine computational hardness. I now turn my attention to how computational requirements have been studied. In other words, I present what TCC maps *into*. In this framework, the requirements are studied by considering the average computational requirements of solving a *random ensemble* of instances. This involves defining a specific random instance generation process, and thus, does not characterize computational hardness of a specific instance, but instead the expected hardness of a random ensemble.

Computational requirements have been predominately studied by looking at the average time-complexity of instances (Cheeseman, Kanefsky, and W. M. Taylor 1991; D. Mitchell, Selman, and Levesque 1992; Ian P Gent and Walsh 1996; Nudelman et al. 2004; Selman and Kirkpatrick 1996; Ian P. Gent et al. 1996). Two general approaches have been used in this regard. One considers exact algorithms and the other considers heuristics. Firstly, the time-complexity has been experimentally studied by implementing exact algorithms and estimating the solve time or number of computational steps of different algorithms. This has been applied to the Boolean satisfiability problem (D. Mitchell, Selman, and Levesque 1992; Selman and Kirkpatrick 1996), the TSP (Ian P Gent and Walsh 1996), the knapsack problem (Yadav et al. 2020) among many others (Cheeseman, Kanefsky, and W. M. Taylor 1991; Zdeborová and Mézard 2008). Secondly, the computational requirements have been calculated by analyzing how well particular heuristics perform on instances with

varying levels of constrainedness (Zdeborová and Mézard 2008; Krzakala and Zdeborová 2007).

Overall, the objective has been to characterize levels of constrainedness (α) at which the hardest instances might be found independent of the solver. This approach allows for the estimation of the average hardness of an algorithm (or a set of them) for instances with the same level of constrainedness (α). Critically, however, the values of α at with heuristics typically break, and at which exact algorithms require more computational resources, has been shown to be consistent across solvers. Therefore, it has been suggested that computational hardness as characterized here is an intrinsic property of an instance.

* * *

In summary, the line of research just presented has identified a set of structural properties of the problem that predict computational hardness across problems and algorithms. The mathematical properties are characterized by the constrainedness parameter (α) and the related threshold values at which the structure of the problem changes. Both constrainedness and thresholds have been shown to be linked to the computational requirements (or reliability) of a plethora of algorithms implemented by electronic computers. It remains an open question whether this framework can be applied to the study of human computation. In the next chapter, I present a proposal for how to operationalize this mathematical framework so that it can be applied to the study of cognition.

Chapter 2

Theory: Typical-case complexity in cognition

Typical-case complexity (TCC) maps a set of mathematical properties of an instance of a problem (α) to the expected computational hardness of solving a randomly sampled instance. This mapping, however, is theoretically defined in the limit as the size of the instance tends to infinity ($n \to \infty$). This is problematic for the application of this theory to human cognition because the size of the instances humans solve might be small enough so that the asymptotic properties do not apply.

In this chapter, I present an explicit definition of TCC based on how computational hardness has been studied in the literature, but that is capable of characterizing hardness for finite values of n. Afterwards, I define *instance complexity* (IC), a related metric that captures the computational hardness of a single instance without reference to a *random ensemble* (that is, a collection of randomly sampled instances). I show that both metrics indeed converge to the canonical asymptotic definition of computational hardness from the literature on random ensembles. Finally, I propose a pipeline to apply this framework to new computational problems.

2.1 TCC Definition

TCC has been defined for constraint satisfaction problems. These problems are ubiquitous in real life, including problems such as the knapsack decision problem (Fig 1.2) as well as variants of the traveling salesperson problem and the tower of London problem. Overall, these problems are defined as those in which the aim is to determine whether there exists a state for which a set of objects satisfy a number of constraints. Formally,

Definition 2.1. *Constraint satisfaction problem. A constraint satisfaction problem consists of a triplet* $\{X, D, C\}$ *where* $X = \{X_1, \dots X_n\}$ *is a set of variables and* $D = \{D_1, \dots, D_n\}$ *is a set of domains of values that the respective variables can take.*

$C = \{C_1, \dots, C_m\}$ *is a set of constraints. Each constraint* C_i *is a pair* $(\bar{X}_i, R_i)$ *where* $\bar{X}_i \subseteq X$ *is a subset of* k *variables in* X *and* R_i *is a* k*-ary relation on the corresponding subset of domains* $\bar{D}_i$*.*

The objective is to find whether there exists a value assignment $g : X \mapsto D$ *that maps all of the variables into their respective domain such that all constraints* C *are satisfied.*

To illustrate, in the 0-1 knapsack decision problem the variables correspond to the items and the constraints correspond to the capacity and the target profit. The domain of each variable is $D_i = \{0, 1\}$ since each item can only be either in (1) or out (0) of the knapsack. The question in this problem is then to assess whether there exists an assignment of items X to be in or out of the backpack such that all constraints (C) are satisfied.

Constraint satisfaction problems are a type of decision problem and, as such, their answer is either 'yes' or 'no'. Instances whose answer is 'yes' are called *satisfiable* instances, otherwise they are called *unsatisfiable*. Note that to verify that the instance is satisfiable, it suffices to find a single value assignment (g) that satisfies the constraints (C). These assignments are denoted *witnesses* of the solution since they verify the solution. Importantly, there might be many such assignments:

Definition 2.2. *Number of witnesses. The number of witnesses η of a particular instance I of a constraint satisfaction problem is defined as the number of assignments g that satisfy the constraints C.*

This mathematical property of an instance is the basis from which to define constrainedness (α). This parameter is set to capture how lenient or restrictive the constraints are for a fixed input size n. Importantly, this is done in expectation. That is, constrainedness characterizes the leniency of the constraints for a random ensemble of instances. This random ensemble is generated by a random process ψ that generates a specification of an *instance* $I(\psi)$ of a problem under certain overarching structural boundaries Π, such that $I \in \Pi$. Π characterizes the problem. In the case of the knapsack problem, the structural boundaries Π of the problem stem from the definition of the problem, which includes the definition of the domains of the variables $X_i \in \{0, 1\}$ as well as the constraint definitions with regards to the target profit and the knapsack capacity. On the other hand, an instance can be defined as the particular weights and values of the items together with the specific target capacity and profit values.

Constrainedness is defined based on the underlying stochastic properties of a random ensemble. Explicitly, it has been defined as a function of the properties of the instance that determine the *satisfiability probability* (Cheeseman, Kanefsky, and W. M. Taylor 1991) and the expected number of solution witnesses (Ian P. Gent et al. 1996). Here, I define constrainedness based on the expected number of solutions, which is defined as the stochastic counterpart of the number of solution witnesses for a random ensemble.

Definition 2.3. *Constrainedness (α). Let $I(\psi) \in \Pi$ be a random instance of a problem Π and a random sampling process ψ. Let $\eta_\psi : \Pi \to \mathbb{N}$ be a function that maps an instance I of a problem to the expected number of witnesses $\eta_\psi(I)$, given a random sampling process ψ. The constrainedness parameter(s) α corresponds to a mapping ($\alpha : \Pi \to \mathbb{R}^k$) from instances of a problem to k parameters ($\alpha(I) \in \mathbb{R}^k$) such that there exists a mapping $\bar{\eta}_\psi : \mathbb{R}^k \mapsto \mathbb{N}$ that satisfies*

$$\bar{\eta}_\psi(\alpha(I)) = \eta_\psi(I) \quad \forall I(\psi) \in \Pi$$

In other words, α is a mapping that generates a set of parameters that encode all the relevant information of the instance needed to determine the expected number of solutions. Note that α is a mapping from instances, which might depend on

the random generation process. However, to simplify notation, I will refer to this parameter as $\alpha(I)$ or even α.

Based on the definition of constrainedness, I now turn to defining the TCC mapping:

Definition 2.4. *TCC. Let $I(\psi) \in \Pi$ be a random instance of a problem Π and a random sampling process ψ. Let $\alpha(I) \in \mathbb{R}^k$ be the constrainedness parameter(s). TCC is defined as a mapping $TCC : \mathbb{R}^k \to \mathbb{R}$ from the constrainedness of an instance $\alpha(I)$ to its expected hardness:*

$$TCC_{\psi}(\alpha(I)) = d(\alpha(I), \alpha^{thr}(\psi))$$

where $d(\cdot)$ is a distance function and α^{thr} is a threshold value(s) of α.

Recent studies have found a number of asymptotic thresholds that are related to the hardness of an instance (Krzakala and Zdeborová 2007; Ardelius and Zdeborová 2008; Remi Monasson et al. 1999). Prominently, a satisfiability threshold (α^{sat}) has been related to an increase in the computational requirements of solving an instance (Cheeseman, Kanefsky, and W. M. Taylor 1991; D. Mitchell, Selman, and Levesque 1992; Ian P Gent and Walsh 1996; Nudelman et al. 2004; Selman and Kirkpatrick 1996; Ian P. Gent et al. 1996). This threshold, however, is defined asymptotically as $n \to \infty$ and it is not clear how this definition can be extended to finite values of the input size. Here, I present one such generalization.

Definition 2.5. *Satisfiability threshold. α^{sat} is defined as the value(s) of α at which the satisfiabilty probability is 0.5 for a given input size (n). Explicitly,*

$$\alpha^{sat}(\psi) = \alpha\Big|_{P_{\psi}(satisfiable|\alpha)=0.5}$$

In this chapter, and in this thesis more generally, I focus my attention on the satisfiability threshold. Explicitly, I define $TCC_{\psi}(\alpha)$ with regards to the distance between α and the satisfiability threshold (α^{sat}). Here, I also define the distance function as euclidean distance (i.e., absolute value for $k = 1$) between α and $\alpha^{sat}(\psi)$.

Definition 2.6. *$TCC^{sat}_{\|\cdot\|}$. In this thesis, TCC is defined as the $TCC(\alpha)$ metric for which*

$$\alpha^{thr}(\psi) = \alpha^{sat}(\psi)$$
$$d(\alpha, \alpha^{thr}) = \|\alpha - \alpha^{thr}\|_2$$

2.2 Instance complexity

The previous section presented a metric of expected hardness of a random instance based on the theoretical framework encompassing typical-case complexity (TCC). It remains an open question whether this framework can be extended to capture hardness of a particular instance of the problem. In this section, I introduce an alternative extension of TCC to finite values of the input size ($n << \infty$). This new metric can capture hardness at a more granular level and avoid the need for the specification of a random sampling process.

As mentioned before, TCC has been canonically studied as $n \to \infty$. Under the assumption of a phase transition in the limit, there are at least two alternative explanations of how constrainedness maps into hardness at finite values of the input size while coinciding with the work on asymptotic properties. Firstly, as with the definition of TCC, the expected hardness of an instance could be related to the proximity of the constrainedness of an instance to the satisfiability threshold. Alternatively, hardness could be driven by how far an instance's constrainedness is to the maximum level of constrainedness for which the instance would be satisfiable. The previous section explored the former alternative, while this section introduces the latter.

I first define the maximum satisfiable constrainedness for a single instance and then use this concept to define *instance complexity* (IC).

Definition 2.7. *Maximum satisfiable constrainedness (α^*). Let Let $I \in \Pi$ be an instance of a problem Π. Let $I'(x)$ be the same instance as I except that the constraints of the problem are modified in such way that $\alpha(I'(x)) = x$. The maximum satisfiable constrainedness of I is defined as*

$$\alpha^*(I) = \max x$$
$$s.t.$$
$$I'(x) \text{ is satisfiable}$$
$$I'(x+\epsilon) \text{ is NOT satisfiable } \forall \epsilon > 0$$

The definition of α^* is for a single instance and does not require the definition of a random sampling process. This provides a natural way to define a metric of instance hardness that is specific to a single instance:

Definition 2.8. *Instance Complexity (IC). Let $I \in \Pi$ be an instance of a problem Π. Let $\alpha(I) \in \mathbb{R}^k$ be the constrainedness parameter(s) and α^* the maximum satisfiable constrainedness. $IC : \Pi \to \mathbb{R}$ is defined as*

$$IC_\alpha(I) = d(\alpha(I), \alpha^*(I))$$

where $d(\cdot)$ is a distance function.

Note that IC might depend indirectly on the sampling procedure because it is defined for a particular $\alpha(\cdot)$. In this chapter and, overall in this manuscript, I refer to this mapping as $IC(I)$ and restrict my analysis to the cases in which the distance function is defined as the Euclidean distance and $\alpha \in \mathbb{R}$.

2.3 TCC, IC and asymptotic definitions

In this section, I explore the link between the definitions presented here of TCC and IC, with previous definitions of typical-case complexity based on the study of random ensembles. To do this, I define the *expected instance complexity* (EIC), the stochastic counterpart of IC for random ensembles. Employing this definition, I show that EIC and TCC both converge to the standard asymptotic definition of average hardness from the literature on random ensembles ($\widetilde{TCC}_\infty$).

The classical definition of hardness in this framework presumes the existence of a phase transition in the satisfiability probability as $n \to \infty$. Explicitly, this phase transition can be defined as follows:

Definition 2.9. *Phase Transition. A satisfiability probability that undergoes a phase transition is defined as a convergence in distribution in which*

$$\lim_{n\to\infty} P_{\psi_n}(satisfiable|\alpha) = P_{\psi_\infty}(sat|\alpha) = \begin{cases} 1 & if\ \alpha \leq \alpha^{sat}_{\psi_\infty} \\ 0 & otherwise \end{cases}$$

where ψ_∞ represents the latent random generation process $lim_{n\to\infty}\psi_n$ that is represented by the satisfiability probability described.

The classical definition of hardness with respect to this phase transition can then be described as:

Definition 2.10. $\widetilde{TCC}_\infty$. *Assuming the existence of a phase transition in the satisfiability probability, the mapping $\widetilde{TCC}_\infty : \mathbb{R} \to \mathbb{R}$ is defined as*

$$\widetilde{TCC}_\infty(\alpha) = |\alpha - \alpha^{sat}_{\psi_\infty}|$$

This definition differs from the previous definition of TCC in that the threshold in $\widetilde{TCC}_\infty$ is considered to be a constant value regardless of the value of n. This hinders its applicability to small values of n where the satisfiability probability does not undergo a phase transition and where $\alpha^{sat}_{\psi_\infty}$ and $\alpha^{sat}_{\psi_n}$ might differ significantly.

Note that $\widetilde{TCC}_\infty$ and TCC are defined over random ensembles while IC is not. In order to compare the different definitions, I present first an extension of IC to random ensembles.

Definition 2.11. *Expected Instance Complexity (EIC). Let $I(\psi) \in \Pi$ be a random instance of a problem Π generated by a random sampling process ψ. Let $\alpha(I) \in \mathbb{R}^k$ be the constrainedness parameter(s) and α^* the maximum satisfiable constrainedness. $EIC : \mathbb{R}^k \to \mathbb{R}$ is defined as*

$$EIC_\psi(\alpha(I)) = E_\psi[d(\alpha(I), \alpha^*_\psi)]$$

where $d(\cdot)$ is a distance function and α^*_ψ is the expected maximum satisfiable constrainedness for the sampling process ψ.

As stated before, I assume $k = 1$ and that $d(x, y) = |x - y|$. In what follows, I explore the relation between these alternative definitions of computational hardness over random ensembles. To do this, I start by estimating the expected maximum satisfiable constrainedness $E(\alpha^*)$ in the limit.

Lemma 2.1. $E_\infty(\alpha^*)$. *Let ψ be random generation process of instances $I(\psi)$ of a problem Π. Let the satisfiability probability undergo a phase transition as $n \to \infty$ and let $\lim_{n\to\infty} \alpha^{sat}_{\psi_n} = \alpha^{sat}_{\psi_\infty}$. Under this probability distribution:*

$$\alpha^*_{\psi_\infty} = \alpha^{sat}_{\psi_\infty}$$

Proof. This follows directly from the definition of α^*. Under the probability distribution $P_{\psi_\infty}(\text{satisfiable}|\alpha)$, the only value of α for which $I'(\alpha)$ is satisfiable and $I'(\alpha + \epsilon)$ is NOT satisfiable $\forall\epsilon > 0$ is α^{sat}_∞. □

Lemma 2.2. *$\widetilde{TCC}_\infty$, TCC_∞ and EIC_∞. Let ψ_n be a random generation process of instances $I(\psi_n)$ of a problem Π. Let $d(a,b) = |a-b|$ be the Euclidean metric. Under the assumptions of Lemma 2.1, $\widetilde{TCC}_\infty$, TCC and EIC converge as $n \to \infty$:*

$$\lim_{n\to\infty} TCC_{\psi_n}(\alpha) = \lim_{n\to\infty} EIC_{\psi_n}(\alpha) = \widetilde{TCC}_\infty(\alpha)$$

Proof. The asymptotic value of TCC can be readily estimated based on the assumptions presented in Lemma 2.1:

$$\begin{aligned}\lim_{n\to\infty} TCC_{\psi_n}(\alpha) &= \lim_{n\to\infty} |\alpha - \alpha^{sat}_{\psi_n}| \\ &= |\alpha - \lim_{n\to\infty} \alpha^{sat}_{\psi_n}| \\ &= |\alpha - \alpha^{sat}_{\psi_\infty}|\end{aligned}$$

I now turn my attention to the asymptotic value of EIC. Let $M(\alpha)$ be the cumulative distribution function of the maximum satisfiable constrainedness. This can be related to the satisfiability probability as follows:

$$\begin{aligned}M(\alpha) = P(\alpha^* \leq \alpha) &= P(\text{unsatisfiable}|\alpha(I) = \alpha) \\ &= 1 - P(\text{satisfiable}|\alpha(I) = \alpha)\end{aligned} \tag{2.1}$$

By this equation, it is possible to conclude that the distribution of the random variable α^*_n converges as well to a corresponding distribution M_{ψ_∞}:

$$\lim_{n\to\infty} M_{\psi_n}(\alpha^*) = 1 - \lim_{n\to\infty} P_{\psi_n}(sat|\alpha = \alpha^*) = 1 - P_{\psi_\infty}(sat|\alpha = \alpha^*) = M_{\psi_\infty}$$

Under the additional assumption that the sequence of random variables $\{\alpha_n\}_{n\in\mathbb{N}}$ are uniformly integrable, we have:

$$\begin{aligned}\lim_{n\to\infty} EIC_{\psi_n}(\alpha) &= \lim_{n\to\infty} E_{\psi_n}(|\alpha - \alpha^*|) \\ &= E_{\psi_\infty}(|\alpha - \alpha^*|) \\ &= E_{\psi_\infty}(\alpha - \alpha^*|\alpha^* < \alpha)P_{\psi_\infty}(\alpha^* < \alpha) + E_{\psi_\infty}(\alpha^* - \alpha|\alpha^* \geq \alpha)P_{\psi_\infty}(\alpha^* \geq \alpha) \\ &= (\alpha - \alpha^*_{\psi_\infty})P_{\psi_\infty}(\alpha^* < \alpha) + (\alpha^*_{\psi_\infty} - \alpha)P_{\psi_\infty}(\alpha^* \geq \alpha) \\ &= \begin{cases} \alpha - \alpha^*_{\psi_\infty} & \text{if } \alpha > \alpha^*_{\psi_\infty} \\ \alpha^*_{\psi_\infty} - \alpha & \text{if } \alpha \leq \alpha^*_{\psi_\infty} \end{cases} \\ &= |\alpha - \alpha^*_{\psi_\infty}|\end{aligned}$$

Note that the assumption of uniform integrability can be met by verifying any single one of the following conditions[1]:

1. $\lim_{a\to\infty} \sup_n E(|\alpha^*_{\psi_n}|1_{|\alpha^*_{\psi_n}|>a}) = 0$
2. $\exists b, N$ s.t. $P_{\psi_n}(|\alpha^*| \geq b) = 1\ \forall n \geq N$
3. $\exists c < d$ and N s.t. $P_{\psi_n}(sat|\alpha = c) = 1$ and $P_{\psi_n}(sat|\alpha = d) = 0\ \forall n \geq N$

[1]These conditions, and especially condition 3, are fulfilled and easily verified on many problems. In particular, this is the case for the TSP, the Boolean satisfiability problem and the 0-1 knapsack decision problem.

Alternatively, the uniform integrability assumption is not necessary if the distance metric is transformed to a bounded function such that $\exists M \in \mathbb{R}$ s.t. $d(a,b) < M \ \forall a, b \in \mathbb{R}$.

Finally, by Lemma 2.1, it follows that if the satisfiability probability undergoes a phase transition then

$$\lim_{n\to\infty} EIC_{\psi_n}(\alpha) = |\alpha - \alpha^*_{\psi_\infty}| = |\alpha - \alpha^{sat}_{\psi_\infty}| = \lim_{n\to\infty} TCC_{\psi_n}(\alpha)$$

□

Note that, although IC is not defined on a random ensemble, it is possible to analyze its asymptotic behavior by considering a sequence of instances randomly generated with a fixed value of $\bar{\alpha}$ and increasing size:

$$\{I_n\}_{n\in\mathbb{N}} \text{ s.t. } \alpha(I_n) = \bar{\alpha}$$

For this sequence of instances, it is easy to verify that IC would converge to both TCC and EIC:

$$\lim_{n\to\infty} IC(I_n) = |\alpha - \alpha^*_{\psi_\infty}|$$

The results presented so far in this section suggest that asymptotic typical-case complexity as it is canonically studied in computer science, can stem from (at least) two different sources, namely from EIC and/or TCC. However, these convergence results hold only asymptotically and under the assumption that the satisfiability probability undergoes a phase transition. This raises the question about how TCC and EIC are linked to each other for a finite values of n where the satisfiability probability does not undergo a phase transition. In what follows, I link EIC to the satisfiability probability and compare these results to the definition of TCC.

EIC derivation for smooth satisfiability probabilities

Recall from equation 2.1 that

$$M(\alpha) = 1 - P(\text{satisfiable}|\alpha(I) = \alpha)$$

where $M(\alpha)$ is the cumulative distribution function of the maximum satisfiable constrainedness. This implies that the probability of the maximum satisfiable constrainedness lying in a range of values of α is directly related to the slope of the satisfiability probability:

$$P(a \leq \alpha^* \leq b) = P(\text{satisfiable}|\alpha(I) = a) - P(\text{satisfiable}|\alpha(I) = b)$$

If the satisfiability probability is continuously differentiable with respect to α, then the probability density function of the maximum satisfiable constrainedness corresponds to the derivative of the satisfiability probability:

$$\begin{aligned} m(\alpha) &= \frac{dM(\alpha)}{d\alpha} \\ &= -\frac{dP(\text{satisfiable}|\alpha(I) = \alpha)}{d\alpha} \end{aligned}$$

Therefore, the expected maximum satisfiable constrainedness $EIC_\psi(\alpha)$ will directly depend on the the derivative of the satisfiability probability:

$$
\begin{aligned}
EIC_\psi(\bar{\alpha}) &= \int_{\mathbb{R}} |\bar{\alpha} - \alpha^*| \; m(\alpha^*) d\alpha^* \\
&= \int_{-\infty}^{\bar{\alpha}} (\bar{\alpha} - \alpha^*) \; m(\alpha^*) d\alpha^* + \int_{\bar{\alpha}}^{\infty} (\alpha^* - \bar{\alpha}) \; m(\alpha^*) d\alpha^* \\
&= \int_{-\infty}^{\bar{\alpha}} \bar{\alpha} \; m(\alpha^*) d\alpha^* - \int_{\bar{\alpha}}^{\infty} \bar{\alpha} \; m(\alpha^*) d\alpha^* + \int_{\bar{\alpha}}^{\infty} \alpha^* \; m(\alpha^*) d\alpha^* - \int_{-\infty}^{\bar{\alpha}} \alpha^* \; m(\alpha^*) d\alpha^* \\
&= \bar{\alpha}[M(\bar{\alpha}) - (1 - M(\bar{\alpha}))] + \int_{\bar{\alpha}}^{\infty} \alpha^* \; m(\alpha^*) d\alpha^* - \int_{-\infty}^{\bar{\alpha}} \alpha^* \; m(\alpha^*) d\alpha^* \\
&= \bar{\alpha}[2M(\bar{\alpha}) - 1] + \int_{\bar{\alpha}}^{\infty} \alpha^* \; m(\alpha^*) d\alpha^* - \int_{-\infty}^{\bar{\alpha}} \alpha^* \; m(\alpha^*) d\alpha^* \\
&= \bar{\alpha}[2M(\bar{\alpha}) - 1] - \int_{\bar{\alpha}}^{\infty} \alpha^* \; \frac{dP(\text{sat}|\alpha = \alpha^*)}{d\alpha^*} d\alpha^* + \int_{-\infty}^{\bar{\alpha}} \alpha^* \; \frac{dP(\text{sat}|\alpha = \alpha^*)}{d\alpha^*} d\alpha^* \\
&= \bar{\alpha}[2(1 - P(\text{sat}|\alpha = \bar{\alpha}) - 1] - \int_{\bar{\alpha}}^{\infty} \alpha^* \; \frac{dP(\text{sat}|\alpha^*)}{d\alpha^*} d\alpha^* + \int_{-\infty}^{\bar{\alpha}} \alpha^* \; \frac{dP(\text{sat}|\alpha^*)}{d\alpha^*} d\alpha^* \\
&= \bar{\alpha}[1 - 2P(\text{sat}|\alpha = \bar{\alpha})] - \int_{\bar{\alpha}}^{\infty} \alpha^* \; \frac{dP(\text{sat}|\alpha^*)}{d\alpha^*} d\alpha^* + \int_{-\infty}^{\bar{\alpha}} \alpha^* \; \frac{dP(\text{sat}|\alpha^*)}{d\alpha^*} d\alpha^*
\end{aligned}
$$

The results from this section suggest a set of different predictions from EIC and TCC. Critically, while TCC does not predict an effect of the slope of the satisfiability probability on hardness, EIC does.

* * *

Overall, TCC and IC might jointly provide a valuable framework for the study of cognition. Firstly, both can be employed to investigate the effect of computational hardness on human performance and effort. Secondly, TCC can be used to characterize subjective beliefs of task difficulty. Indeed, TCC is a metric that can be potentially employed by agents to generate subjective beliefs of hardness of a task given that it can be estimated from the features of the task without the need to solve the problem. In the next section, I propose a pipeline to generate these metrics for new problems for the study of human computation.

2.4 A pipeline for new computational problems in this framework

In the previous section, I introduced a generic approach, based on the constrainedness parameter, to study hardness of instances in cognition. This approach is generic because it presumes that problems have an intrinsic hardness, regardless of the strategy used to solve the problem. Importantly, it is also generic in the sense that it is problem-independent. It can potentially be applied to any computational decision problem to characterize hardness of random ensembles and individual instances. I propose the following framework as a pipeline to characterize the computational hardness, related to constrainedness, of a computational problem:

1. Specify a random sampling process for instances of the problem.
2. Estimate, analytically, the expected number of solution witnesses of the problem.
3. Based on the expected number of solution witnesses, characterize the constrainedness parameter(s) α.
4. Characterize, via simulations, the satisfiability threshold (α^{sat}) for the relevant input-size (n).

The pipeline can be simplified if an analytical expression for the satisfiability probability is characterized:

1. Specify a random sampling process for instances of the problem.
2. Estimate, analytically, the satisfiability probability of the problem.
3. Based on the satisfiability probability, characterize the constrainedness parameter(s) α and the satisfiability threshold (α^{sat}).

In the next chapter, I apply this pipeline to characterize hardness and test its effects on human decision quality in the knapsack problem.

Chapter 3

The Knapsack Case

In this chapter I present the co-authored paper titled *"Generic properties of a computational task predict human effort and performance"*. There we apply the proposed framework from the previous chapter to the knapsack problem and investigate the effect of the corresponding metrics of computational hardness on human performance. Moreover, we provide a generalization of the TCC metric to optimization problems (TCC_O) and empirically test its ability to predict human behavior.

Generic properties of a computational task predict human effort and performance

Juan Pablo Franco, Nitin Yadav, Peter Bossaerts, Carsten Murawski

Abstract

It has been shown that computational hardness of cognitive tasks affects people's effort and ability to solve problems reliably. However, prior empirical studies lack generality. They quantify computational hardness of tasks based on particular algorithms or for specific problems. Here, we propose a set of measures of computational hardness of individual instances of a task in a way that is independent of any algorithm or computational model and can be generalized to other problems. Specifically, we introduce two measures, typical-case complexity (TCC), a measure of average hardness of a random ensemble of instances, and instance complexity (IC), an instance-specific metric. Both measures are related to structural properties of instances. We then test the effect of those measures on human behavior by asking participants to solve instances of two variants of the 0-1 knapsack problem, a canonical and ubiquitous NP-hard problem. We find that participants spent more time on instances with higher TCC and IC, but that decision quality was lower in those instances. We propose that the study of mathematical properties of tasks related to computational hardness can contribute to the development of computationally plausible accounts of human decision-making, just like stochastic properties have proven to be critical to our understanding of human decisions in probabilistic tasks.

3.1 Introduction

Life requires us to make complex decisions with limited cognitive resources. Theories of human cognition and behavior such as bounded rationality (Gerd. Gigerenzer and Selten 2001; Gerd Gigerenzer and Brighton 2009; Herbert A Simon 1990), the heuristics and biases approach (Kahneman and Tversky 1979; Tversky and Kahneman 1992) and others take these cognitive limitations into account, either explicitly or implicitly. However, a complete account of human cognition and decision-making needs to account not only for the limited cognitive capacities of people but also for the cognitive demands of the tasks people face. Here, we provide a framework for studying the latter.

In this framework, the heuristics that people have been observed to use (Kahneman and Tversky 1979; Tversky and Kahneman 1992) may emerge as the very strategies that a rational agent would be expected to use to overcome the computational overload imposed by some of the computational problems faced. That may sound self-evident, but it is not once it is appreciated that the use of a particular heuristic is not only driven by an agent's cognitive limitations but by the interaction of these limitations and the cognitive resource requirements imposed by the particular instance of a cognitive task.

A key question in the study of the interaction between task requirements and cognitive limitations is whether there exist properties of individual instances of problems that make them computationally hard. Several approaches have studied hardness of instances and their effect on human performance. Prominently, it has been shown that algorithm-specific metrics of computational hardness predict human effort and performance in cognitive tasks including decisions (e.g., Acuña and Parada 2010; Murawski and Peter Bossaerts 2016; Guid and Bratko 2013; De Visscher and Noël 2014; MacGregor and Chu 2011). For instance, a metric of difficulty based on an extension of the greedy algorithm (Sahni-k algorithm; Sahni and Sartaj 1975) has been suggested to quantify hardness and its effect on human performance in the knapsack problem (Murawski and Peter Bossaerts 2016). This approach is problematic, though, because these metrics assume that the agent follows a specific computational strategy. Overall, this program ignores the diversity in strategies used by humans (e.g., MacGregor and Chu 2011; Acuña and Parada 2010; Hirtle and Gärling 1992; Ohlsson 2012; Gerd Gigerenzer and Gaissmaier 2011; B. R. Newell, Weston, and Shanks 2003; Payne, Bettman, and Johnson 1993) and overlooks the difficulty of identifying such strategies.

The question that arises then is whether there exist properties of individual instances related to their computational hardness that are independent of the algorithm used. Previous research has studied the intrinsic hardness of tasks (independent of algorithms) by exploring problem-specific metrics of difficulty (e.g., MacGregor and Chu 2011; Hirtle and Gärling 1992; Kotovsky, Hayes, and H. A. Simon 1985; Carruthers, Masson, and Stege 2012; Bourgin et al. 2017; Shepard and Metzler 1971; Stazyk, Ashcraft, and Hamann 1982). However, the approaches employed in these studies are not readily generalizable to other tasks. For instance, the traveling salesman problem has been widely studied and sources of difficulty of the problem have been related to the graphical representation of the optimal itinerary (e.g., convexity) and the node distribution (e.g., degrees of clustering; MacGregor and Chu 2011). These metrics, however, are specific to the problem and it is not

clear if they can be modified to elucidate hardness in other problems. Even an alternative representation of the same problem (e.g., numeric distance matrix of the traveling salesman problem) might require a different set of metrics to quantify hardness. Overall, this approach does not reveal which underlying task-independent properties of computational problems make them hard for people to solve.

A desirable property of metrics of complexity is for them to be *generic*, that is, problem and strategy independent. Just like like mean and variance characterize the level of (stochastic) uncertainty in probabilistic tasks, generic metrics of complexity would allow us to characterize computational hardness across tasks. This could shed light on which strategies are employed in problem-solving tasks, similar to how stochastic properties of a task have informed how people approach probabilistic tasks (e.g., Payzan-Lenestour and Peter Bossaerts 2011; Averbeck 2015; Daw et al. 2006). Additionally, generic metrics of hardness could inform how time and effort are allocated in problem-solving tasks. This is similar to how intrinsic properties of perceptual stimuli, such as coherence or evidence strength more generally, have played an important role in the study of effort-accuracy trade-offs in perceptual tasks (e.g., Drugowitsch et al. 2012; Hanks and Summerfield 2017).

One generic metric of hardness in computational problems that has been investigated previously is problem size. Several studies have shown that human performance worsens as the size of the problem increases (e.g., Carruthers, Masson, and Stege 2012; MacGregor and Chu 2011; Dry et al. 2006; van Opheusden and Ma 2019; Stazyk, Ashcraft, and Hamann 1982; De Visscher and Noël 2014). However, this dimension of complexity is unable to account for differences in hardness and performance across instances with the same problem size. It remains an open question whether there are additional generic dimensions of complexity that affect human performance.

Advances in computational complexity theory have related hardness of computational problems to generic mathematical properties of instances of those problems. In particular, the *constrainedness* of an instance has been related to the amount of computational resources algorithms need to find a solution (Arora and Barak 2009; Monasson et al. 1999; Cheeseman, Kanefsky, and Taylor 1991; Ian P. Gent et al. 1996). Importantly, the relation between instance properties and computational hardness is independent of the algorithm used. This work suggests that computational hardness of instances can be specified as an intrinsic property of instances.

The question addressed in the present study is to what extent these instance-level measures of computational hardness affect human ability to solve an instance. The answer to this question is not obvious (Blakey 2011). Firstly, we do not currently have a model of human computation and do not know in which ways human computation differs from other models of computation, such as a Turing machine or a quantum computer (Blum and Vempala 2020). Secondly, and relatedly, we cannot directly observe which algorithm, if any, a person uses to solve a computational problem.

Here, we investigated how the mathematical structure of individual instances of the 0-1 knapsack problem affects human decision quality and time-on-task. The knapsack problem is a canonical NP-hard computational problem that is closely related to many theories of decision-making such as utility maximization (Von Neumann and Morgenstern 1947) and satisficing (Herbert A Simon 1956). Specifically, it describes a multi-attribute decision problem with two (conflicting) attributes (weight

and value). However, its relevance extends beyond decision theory. The problem manifests itself in many tasks faced in everyday life such as to choice of which stimuli to attend to, budgeting and time management, portfolio optimization, intellectual discovery as well as in industrial applications such as the cargo business (Kellerer, Pferschy, and Pisinger 2004; Meloso, Copic, and Bossaerts 2009).

We propose two metrics that generically capture computational hardness of an instance for algorithms that solve instances reliably. First, we build on the concept of typical-case complexity (TCC), a popular approach for studying intrinsic hardness of random ensembles of instances of NP-hard problems (Cheeseman, Kanefsky, and Taylor 1991; Monasson et al. 1999; Percus, Istrate, and Moore 2006). Importantly, it has been shown that there exist features of instances based on which one can predict the average number of computations needed to compute the solution of an instance (Cheeseman, Kanefsky, and Taylor 1991; Krzakala et al. 2006; Ricci-Tersenghi 2010; Achlioptas, Coja-Oghlan, and Ricci-Tersenghi 2011; Marino, Parisi, and Ricci-Tersenghi 2016; Ricci-Tersenghi, Semerjian, and Zdeborová 2019; Budzynski, Ricci-Tersenghi, and Semerjian 2019) (Fig 3.1a). We conjectured that TCC would predict performance and time-on-task for humans. We tested this using two variants of the knapsack problem.

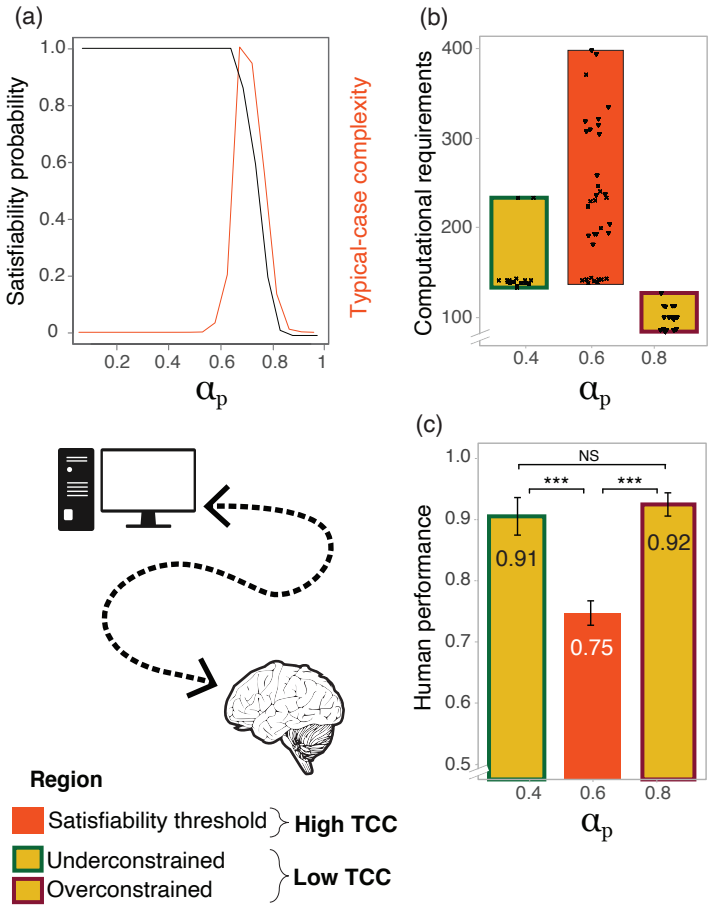
In the second approach, we construct a metric of instance complexity for instances of a decision problem, based on TCC, that is specific to a single instance. We refer to this metric as instance complexity (IC). Computing this metric is more computationally expensive than TCC since it requires solving an optimization problem. But, unlike TCC, it obviates the need to commit to an ensemble of instances and a distribution over this ensemble.

3.2 Materials and Methods

We studied how a set of mathematical properties of random ensembles of instances (TCC) as well as of individual instances (IC) affect human decision quality and time-on-task in two variants of the 0-1 knapsack problem. in both variants, participants were presented with a set of items I with different weights w and values v. In the *decision variant*, participants were asked to decide whether there exists a subset A of items from the set I for which (1) the sum of weights ($\sum_{i\in A} w_i$) is lower or equal to a given capacity c and (2) the sum of values ($\sum_{i\in A} v_i$) is at least as high as a given target profit p. In the related *optimization variant*, participants were asked to select the set of items that maximizes the sum of values ($\sum_{i\in A} v_i$) without exceeding the knapsack's capacity ($\sum_{i\in A} w_i \leq c$). Both variants are NP-hard (and the decision variant is also NP-complete; Kellerer, Pferschy, and Pisinger 2004).

In our study, participants were asked to solve a number of random instances of both variants of the problem. All instances in the experiment had $n = 6$ items. The number of items was selected, based on pilot data, to ensure that the task was neither too difficult nor too easy. Instances varied in their computational complexity.

Figure 3.1: **Typical-case complexity and performance in the knapsack decision task. (a) Computer performance and satisfiability threshold.** Probability of an instance being *satisfiable* as a function of α_p (left axis). The values presented correspond to the knapsack decision problem with 30 items and **fixed** $\alpha_c \approx 0.44$. The right axis shows a pictorial representation of typical-case complexity (TCC; Yadav et al. 2020). **(b) Instance sampling for the behavioral experiment.** Each point is an instance sampled as a function of the proxy for computational requirements (number of propagations using the *Gecode* solver) and normalized profit α_p. **(c) Human performance by typical-case complexity in the knapsack decision task.** Mean performance and standard errors. The values presented in (b) and (c) correspond to the knapsack decision problem with 6 items and **fixed** $\alpha_c \in [0.40 - 0.45]$ (see section 3.2). *Note:* $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$; *NS: not significant.*



3.2.1 Computational complexity

Typical-case complexity

Previous work has studied the connection between instance properties and computational complexity[1]. One prominent approach relates asymptotic characteristics of

[1]In this manuscript we use the broad definition of *computational complexity* to refer to the study of computational resource requirements for solving a task (Arora and Barak 2009; Pudlák 2013; Moore and Mertens 2011). This notion is not to be confused with the definition of computational complexity in terms of complexity classes, in particular, complexity classes based on asymptotic

random instances to typical-case complexity (TCC). This approach has led to the discovery of phase transitions in the solution space (the set of configurations of variables that satisfy the instance's constraints) that have been shown to be related to average hardness (Cheeseman, Kanefsky, and Taylor 1991; Krzakala et al. 2006; Ian P Gent and Walsh 1996; Achlioptas, Naor, and Peres 2005; Selman and Kirkpatrick 1996; Krzakala et al. 2006). Notably, it has been shown for many NP-hard problems that there exists an asymptotic phase transition in the *satisfiability probability* (the probability that the correct answer to the instance is 'yes'), that is, an abrupt jump from one to zero at a particular value of an order parameter. It has been shown that instances with an order parameter closer to the phase transition tend to be harder.

Recent work studied TCC of random instances of the 0-1 knapsack problem (Yadav et al. 2020). They demonstrated that the hardest instances in the decision variant tend to appear in the vicinity of the so-called *satisfiability threshold*, where the probability of an instance being *satisfiable* is close to 0.5. The satisfiability threshold separates instances of the problem into two regions: an under-constrained region where the constraints are lenient, and thus many solutions are likely to exist, and an over-constrained region where the constraints are stringent, and thus the existence of a solution is unlikely (that is, an instance is not satisfiable). Computing the solution of instances in the proximity of the satisfiability threshold requires on average more computational resources than for instances further away from it (Fig 3.1a).

The probability that an instance of the knapsack decision problem is satisfiable can be expressed in terms of a small set of instance parameters $\boldsymbol{\alpha} = (\alpha_p, \alpha_c)$, which we will refer to as normalized profit and normalized capacity, respectively. We define them as follows:

$$\alpha_p = \frac{p}{\sum_{i=1}^{n} v_i} \quad \text{and} \quad \alpha_c = \frac{c}{\sum_{i=1}^{n} w_i}, \tag{3.1}$$

where w_i are the weights of the items, v_i are values of the items, n is the number of items, c is the weight capacity and p is the target profit. Similar to what has been shown in relation to a number of other NP-hard problems (Cheeseman, Kanefsky, and Taylor 1991; Krzakala et al. 2006; Ian P Gent and Walsh 1996; Yadav et al. 2020; Achlioptas, Naor, and Peres 2005; Selman and Kirkpatrick 1996; Krzakala et al. 2006), there exists a mapping from instance properties (α_p and α_c) to computational complexity of the instance. Explicitly, this mapping can be characterized by a distance metric d between $\boldsymbol{\alpha} = (\alpha_p, \alpha_c)$ and a satisfiability boundary $\boldsymbol{\alpha^s}$ where the satisfiabilty probability sP is 0.5; that is, $\boldsymbol{\alpha^s} = \{\boldsymbol{\alpha} | sP(\boldsymbol{\alpha}) = 0.5\}$:

$$TCC(\boldsymbol{\alpha}) = d(\boldsymbol{\alpha}, \boldsymbol{\alpha^s}). \tag{3.2}$$

In order to operationalize this distance metric whilst minimizing assumptions and maximizing power to detect an effect of TCC, we fixed $\bar{\alpha_c}$ and modulated TCC by varying only α_p:

$$TCC(\alpha_p, \bar{\alpha_c}) = |\alpha_p - \alpha_p^s|. \tag{3.3}$$

The instances in this study were chosen, based on this definition, such that they had different levels of TCC. This was done to remove possible sources of noise whilst increasing the power of the experimental design to identify the effect of TCC on performance. TCC is a metric of average hardness, and thus we expected high

worst-case analysis such as P and NP.

variability in performance across instances with the same level of TCC (Fig 3.1b). Given that our study is the first to study the effect of TCC on human performance, and in order to ensure that our experimental design had maximal power to reliably identify this effect, we selected instances from only two levels of TCC. To do this, we chose instances from three discrete levels of constrainedness by varying the normalized profit (α_p) while keeping the normalized capacity fixed (α_c). This allowed us to test the effect of TCC, while differentiating it from a monotone relation between constrainedness and performance. More importantly, it avoided possible confounding effects of α_c on performance.

Specifically, we set $\bar{\alpha_c} \approx 0.425$ and estimated, via simulations, the satisfiability threshold to be $\alpha_p^s \approx 0.625$. Instances were sampled at different distances from the satisfiability threshold α_p^s (Fig 3.1b). Instances near the threshold are categorized as having high typical-case complexity (*high TCC*) whereas instances further away from it—that is, in the under-constrained and over-constrained regions—are categorized as having low typical-case complexity (*low TCC*).

More specifically, in order to select instances for the knapsack decision task, we first fixed the normalized capacity ($\alpha_c \in [0.40, 0.45]$). We then chose the target profit such that the normalized profit corresponded to one of three regions: under-constrained ($\alpha_p \in [0.35, 0.4]$), satisfiability threshold ($\alpha_p \in [0.6, 0.65]$) and over-constrained ($\alpha_p \in [0.85, 0.9]$). We randomly selected 18 instances from the under-constrained bin and 18 from the over-constrained bin. Additionally, we sampled 18 *satisfiable* instances and 18 *unsatisfiable* instances near the satisfiability threshold ($\alpha_p \in [0.6, 0.65]$). Throughout, we ensured that no weight/value combinations were sampled twice. In order to also ensure enough variability between instances in instances near the satisfiability threshold (high TCC), we added a constraint in the sampling. We forced half of the instances close to the satisfiability threshold to have high computational requirements (top 50%), according to an algorithm-specific ex-post complexity measure of a widely-used algorithm (*Gecode*; Gecode Team 2006). Analogously, the other half was selected to have low computational requirements (bottom 50%). More detail is provided in Appendix A.

The characterization of complexity using TCC is based on the satisfiability probability and, therefore, in principle only applicable to decision problems. One can envisage, however, a way in which TCC applies to optimization problems as well, by framing the optimization problem as a sequence of instances of the decision problem: "Is there another set of items with a higher profit that still satisfies the capacity constraint?" In other words, we can model the search process as selecting a subset of items that satisfy the capacity constraint and then deciding whether there exist other combinations that yields a higher profit while still satisfying the constraint. If the answer is yes, then the agent chooses one such combination and asks the same question again. This process is repeated until the answer is no, which means that the optimum has been reached. We approximate the TCC of an optimization problem by the TCC of the implied instance of the decision problem at the optimum of the optimization problem, that is, the instance of the decision variant with profit threshold equal to the optimal value of the corresponding optimization variant (see appendix C).

To generate instances for the optimization task, a sampling process similar to the one for the decision variant was used. We first selected the same normalized capacity bin we specified for the decision task ($\alpha_c \in [0.4 - 0.45]$). Then, in order

to estimate the normalized profit of the optimization variant, we calculated the optimal set of items $A^* \subseteq A$ for each optimization instance. We then estimated the corresponding optimal sum of values ($p^* = \sum_{i \in A^*} v_i$). The normalized profit was then calculated by dividing the target profit by the sum of values of all of the items ($\alpha_p^* = \frac{p^*}{\sum_{i \in A} v_i}$). The optimization TCC (TCC_O) was defined in the same way as the decision problem TCC employing this newly defined α_p^* as the normalized profit. Twelve (12) instances were selected from the *high* TCC_O region ($\alpha_p^* \in [0.6 - 0.65]$) and six (6) were selected from the *low* TCC_O region ($\alpha_p^* \in [0.85 - 0.9]$). It is worth noting that this process did not generate instances in the under-constrained region ($\alpha_p^* \in [0.35 - 0.4]$).

In order to also ensure enough variability between instances with high TCC_O, we added the same constraint as in the knapsack decision task: we forced half of the instances with high TCC_O to have high computational requirements (top 50%), according to an algorithm-specific ex-post complexity measure of a widely-used algorithm (*Gecode*; Gecode Team 2006). Correspondingly, the other half was forced to have low computational requirements (bottom 50%). We provide more details in appendix A.

Number of witnesses

In the decision variant, an alternative metric of hardness is the *number of witnesses*. This is defined as the number subsets of items that satisfy both profit and capacity constraints.

The number of witnesses can be examined at two different levels of analysis. It can be studied directly as a metric that captures constrainedness (and hardness) of a single satisfiable instance. It can also be studied stochastically by computing the *expected number of witnesses*, which can be interpreted as a metric of *expected* hardness of a random ensemble. The latter approach would map a set of features of instances to expected hardness. This is closely related to the computation of TCC. Specifically, the same features (order parameters) that allow us to compute the satisfiability probability, and thus TCC, also characterize the expected number of witnesses. It has already been shown empirically that the order parameters that characterize the satisfiability probability in the knapsack decision problem are α_p and α_c (Yadav et al. 2020). In appendix D we complement these findings and show, analytically, that the same two parameters characterize the expected number of witnesses (see appendix D). This corroborates that TCC and the expected number of witnesses are both encapsulated in our analysis of α_p and α_c. Moreover, this provides further support for the premise that α_p and α_c fully characterize the constrainedness of the knapsack decision problem.

We will also consider the number of witnesses of a single instance as a complexity measure, which is based entirely on properties of a single instance. This metric can explain differences in constrainedness between satisfiable instances. We would expect instances with a lower number of witnesses to be harder, ceteris paribus. However, this metric cannot explain differences in hardness between unsatisfiable instances given that the number of witness for unsatisfiable instances is zero. To address this issue, we now introduce another metric of hardness of individual instances.

Instance complexity

We define *instance complexity* (IC) as the distance between the level of the profit constraint (target profit) and the maximum value attainable in the corresponding instance of the optimization variant of the 0-1 knapsack problem. Specifically,

$$IC = \left| \frac{p - p^\star}{\sum v_i} \right| = \left| \alpha_p - \alpha_p^\star \right|, \tag{3.4}$$

where p is the target profit of the decision instance and $p^\star$ is the maximum value achievable in the corresponding optimization instance, that is, the instance of the optimization variant with the same set of items I and the same capacity constraint c. α_p and $\alpha_p^\star$ denote the normalized values of target profit and optimum value, respectively. We expect IC to be inversely related to computational complexity.

3.2.2 Experiment

Ethics statement

The experimental protocol was approved by the University of Melbourne Human Research Ethics Committee (Ethics ID 1749616). Written informed consent was obtained from all participants prior to commencement of the experimental sessions. Experiments were performed in accordance with all relevant guidelines and regulations.

Participants

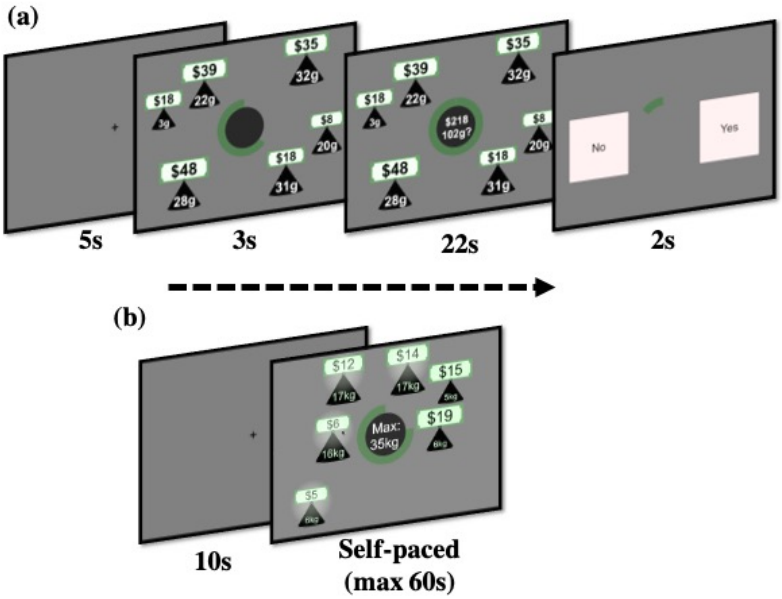
Twenty human volunteers recruited from the general population took part in the study (14 female, 6 male; age range = 18-31 years, mean age = 22.0 years). Inclusion was based on age (minimum = 18 years, maximum = 40 years). Each participant performed the knapsack decision task, the knapsack optimization task, the mental arithmetic task and a set of basic cognitive function tasks.

Knapsack decision task

In this task, participants were asked to solve a number of instances of the (0-1) knapsack decision problem (Fig 3.2a). In each trial, they were shown a set of items with different values and weights as well as a capacity constraint and a target profit. Participants had to decide whether there existed a subset of those items for which (1) the sum of weights is lower or equal to the capacity constraint and (2) the sum of values yields at least the target profit.

Each trial had four stages. In the first stage (3 seconds), only the items were presented. Item values, in dollars, were displayed using dollar bills and weights, in grams, were shown inside a black weight symbol. The larger the value of an item, the larger the dollar bill was in size. Similarly, the larger the weight of an item, the larger its weight symbol was in size. At the center of the screen, a green circle indicated the time remaining in this stage. In the second stage (22 seconds), target profit and capacity constraint were added to the screen inside the green timer circle. In the third stage (2 seconds), participants saw a 'YES' and a 'NO' button on the screen, in addition to the timer circle, and made a response using the keyboard

Figure 3.2: **Knapsack tasks. (a)** Knapsack decision task. Initially, participants were presented with a set of items of different values and weights. The green circle at the center of the screen indicated the time remaining in this stage of the trial. This stage lasted 3 seconds. Then, both capacity constraint and target profit were shown at the center of the screen. Participants had to decide whether there exists a subset of items for which (1) the sum of weights is lower or equal to the capacity constraint and (2) the sum of values yields at least the target profit. This stage lasted 22 seconds. Finally, participants had 2 seconds to make either a 'YES' or 'NO' response using the keyboard. A fixation cross was shown during the inter-trial interval (5 seconds). **(b)** Knapsack optimization task. Participants were presented with a set of items of different values and weights together with a capacity constraint shown at the center of the screen. The green circle at the center of the screen indicated the time remaining in this stage of the trial. Participants had to find the subset of items with the highest total value subject to the capacity constraint. This stage lasted 60 seconds. Participants selected items by clicking on them and had the option of submitting their solution before the time limit was reached. After the time limit was reached or they submitted their solution, a fixation cross was shown for 10 seconds before the next trial started.



(Fig 3.2a). A fixation cross was then shown (5 seconds) before the start of the next trial.

Participants completed 72 trials (3 blocks of 24 trials with a rest period of 60 seconds between blocks). Each trial, a different instance of the knapsack decision problem was presented. The order of instances was randomized across participants.

Knapsack optimization task

In this task, participants were asked to solve a number of instances of the (0-1) knapsack optimization problem (Fig 3.2b). In each trial, they were shown a set of items with different weights and values as well as a capacity constraint. Participants had to find the subset of items that maximized total value subject to the capacity

constraint. This means that while in the knapsack decision problem, participants only needed to determine whether a solution existed, in the knapsack optimization problem, they also needed to determine the nature of the solutions (i.e., the items in the optimal knapsack).

The task had two stages. In the first stage (60 seconds), the items were presented together with the capacity constraint and the timing indicator. Items were presented in the same way as in the knapsack decision task. Unlike in the decision task, however, participants were able to add and remove items to/from the knapsack by clicking on the items. An item added to the knapsack was indicated by a light around it (Fig 3.2b). Participants submitted their solution by pressing the button 'D' on the keyboard before the time limit was reached. If participants did not submit within the time limit, the items selected at the end of the trial were automatically submitted as the solution. Participants were then shown a fixation cross (10 seconds) before the start of the next trial.

Each participant completed 18 trials (2 blocks of 9 trials with a rest period of 60 seconds between blocks). Each trial presented a different instance of the knapsack optimization problem with varying levels of computational complexity. The order of presentation of instances in the task was randomized for each participant.

Basic cognitive function tasks

We tested participants' performance on five aspects of cognitive function that we considered relevant for the knapsack tasks, namely, working memory, episodic memory, strategy use, processing and psychomotor speed, as well as mental arithmetic. To do so, we administered a set of tasks from the Cambridge Neuropsychological Test Automated Battery (CANTAB; see Appendix B; Cognition 2017). Specifically, we asked participants to perform the Reaction Time (RTI), Paired Associates Learning (PAL), Spatial Working Memory (SWM) and Spatial Span (SSP). In addition, participants were presented with 33 mental arithmetic problems (Cappelletti, Butterworth, and Kopelman 2001). The first three trials were considered test trials and thus were not included in the analysis. They were given 13 seconds to solve each problem. The task involved addition and division of numbers, as well as questions in which they were asked to round to the nearest integer the result of an addition or division operation.

Procedure

After reading the plain language statement and providing written informed consent, participants were instructed in the tasks and completed a practice session. Participants first solved the CANTAB RTI task, followed by the knapsack decision task. Then they completed the CANTAB RTI task again, followed by the knapsack optimization task. Subsequently, they completed the remaining CANTAB tasks in the following order: PAL, SWM and SSP. Finally, they performed the mental arithmetic task and completed a set of demographic and debriefing questionnaires. Each experimental session lasted around two hours.

Participants received a show-up fee of A$10, as well as monetary compensation based on performance. They earned A$0.7 for a correct answer in the knapsack decision task and A$1 for a correct answer in the knapsack optimization task.

3.2.3 Statistical analysis

Mixed-effects models were used for the statistical analysis. All of the generalized logistic mixed models (GLMM) and linear mixed models (LMM) included random effects on intercept for participants. Their p-values were calculated using a two-tailed Wald test. All statistical analyses were performed in R and mixed models were estimated using the R package lme4 (Bates et al. 2015).

3.2.4 Data and Code Availability

The raw behavioral data, the data analysis code and the computational simulations are all available at the Open Science Framework. The knapsack decision task, knapsack optimization task and mental arithmetic task are also available there (project: `https://doi.org/10.17605/OSF.IO/T2JV7`).

3.3 Results

We studied how a set of mathematical properties of random ensembles of instances (TCC) as well as of individual instances (IC) affected human decision quality and time-on-task in the decision and optimization variants of the 0-1 knapsack problem.

3.3.1 Knapsack decision task

Summary statistics

We excluded a total of 13 trials (from 8 participants) in which no response was made.

Mean *human performance*, measured as the percentage of trials in which a correct response was made, was 83.1% (min = 0.56, max = 0.9, $SD = 0.08$). On average, participants chose the 'YES' option in 48.1% of trials (min = 0.32, max = 0.60, $SD = 0.06$). Performance did not vary during the course of the task ($P = 0.196$, main effect of trial number on performance, generalized logistic mixed model (GLMM); Table E.6 Model 1), suggesting that neither experience with the task nor mental fatigue affected task performance.

The effect of typical-case complexity (TCC) on performance

In order to test whether participants' ability to solve an instance was affected by TCC, we first compared performance on instances with high TCC and low TCC. We expected participants to perform better on instances with low TCC compared to instances with high TCC. Performance was significantly lower on instances with high TCC ($P < 0.001$, main effect of TCC on performance, GLMM; Fig 3.1c; Table E.6 Model 2).

We hypothesized that performance would be affected by the tightness of the profit and capacity constraints. To examine this, we tested whether performance on instances in the under-constrained region ($\alpha_p \approx 0.4$) was different to performance on instances in the over-constrained region ($\alpha_p \approx 0.9$). We found no significant difference in performance between the two regions with low TCC ($P = 0.355$, main effect of region, GLMM; Table E.6 Model 5; Fig 3.1c), but confirmed a significant

difference in performance between instances with high TCC and each of the other two regions ($P < 0.001$, difference in performance between regions, GLMM; Table E.6 Model 4).

We also hypothesized that the effect of TCC on performance would be affected by the satisfiability of an instance, that is, whether the answer to the decision problem is 'yes' or 'no'. This hypothesis is based on an asymmetry of NP problems. Proving that an instance is *satisfiable* requires finding one subset of items that satisfy the constraints. Such a set may be identified without exploring the full search space and, additionally, there may be more than one such subset. In contrast, to conclude that an instance is *unsatisfiable* requires proving that no such set exists. This might require a full search over every possible subset of items in order to determine that none of the subsets satisfies the constraints. We investigated the effect of satisfiability on performance and found that the effect of TCC was still significant when controlling for satisfiability ($P < 0.001$, main effect of TCC on performance, GLMM; Table E.6 Model 3), but that there was no significant effect of satisfiability on performance ($P = 0.355$ main effect of satisfiability on performance, $P = 0.796$ interaction effect of TCC and satisfiability on performance, GLMM; Table E.6 Model 3).

Structure of an instance and human performance

For satisfiable instances, the tightness of the constraints can be studied further by analyzing the number of solution witnesses. The number of witnesses and TCC are highly related, because both map constrainedness to complexity (see Section 3.2). We tested this link empirically and found, as expected, that satisfiable instances with high TCC tend to have a lower number of witnesses than satisfiable instances with a low TCC ($P < 0.001$, unpaired t-test; Fig 3.3a).

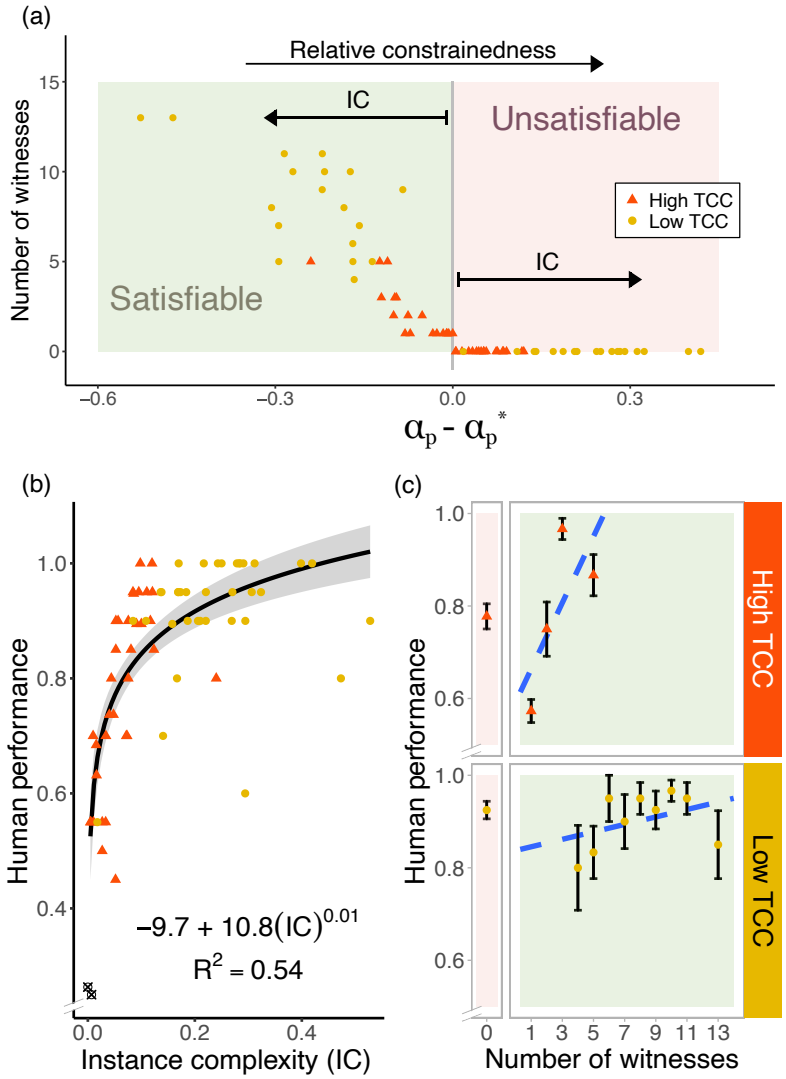
When exploring the link between the number of witnesses and human performance we found that (for satisfiable instances), the probability of solving an instance correctly increased with the number of witnesses (combinations of items that satisfy the constraints; $P = 0.001$, main effect of number of witnesses on performance, GLMM; Table E.6 Model 6; Fig 3.3c). Interestingly, there appeared to be an interaction effect between TCC and number of witnesses on performance: the number of witnesses caused the frequency of correct solutions to increase faster if the instance had high TCC ($P < 0.001$, interaction effect of TCC and number of witnesses on performance; GLMM; Table E.6 Model 6).

Instance complexity (IC) and human performance

We first explored the relation between IC and TCC. Both measures map constrainedness to computational complexity. However, they do so at different levels. TCC maps average constrainedness of a random ensemble of instances to their average complexity, whereas IC maps the constrainedness of a single instance to its complexity, regardless of which ensemble it was sampled from. Given that both measures map constrainedness to complexity, we expected them to be highly correlated. As predicted, instances in our study with *low TCC* had a higher average IC than instances with *high TCC* ($P < 0.001$, $\beta_{TCC} = -0.175$, observations= 70; Fig 3.3b).

We explored the relation between IC and human performance. We found a positive non-linear relation between this measure and average accuracy per instance

Figure 3.3: **Properties of sampled instances and human performance. (a) Properties of sampled instances.** Each decision instance is plotted according to how large the gap between the normalized profit α_p and the maximum achievable normalized profit $\alpha_p^\star$ is, given the set of items and the capacity constraint c. Instances become more constrained as $\alpha_p - \alpha_p^\star$ increases. Instances in the negative quadrant are satisfiable and, thus, have at least one solution witness. The number of witnesses is defined as the number of item combinations that satisfy both capacity and profit constraints. Instances in the positive quadrant are all unsatisfiable and have 0 witnesses. **(b) Relation between IC and human performance in the knapsack decision task.** Mean performance and IC by instance. Instances are divided into high and low TCC. Outliers are denoted in black and excluded from the model fit. **(c) Relation between performance and number of witnesses in the knapsack decision task.** Mean performance and standard error by number of solution witnesses.



($accuracy \sim IC^{0.01}$; $R^2 = 0.542$; best AIC among competing models; Table E.4; Fig 3.3b). We also compared the model fit with respect to a model with only TCC as explanatory variable. As would be expected, IC models performance better than the TCC model ($R^2 = 0.21$; AIC_{TCC} is highest among competing models; Table E.4).

Finally, the effect of IC on performance was further corroborated using a mixed effects model ($P < 0.001$, main effect of $IC^{0.01}$ on performance, GLMM; Table E.6 Model 7).

3.3.2 Knapsack optimization task

Summary statistics

We excluded 2 trials (from 2 participants) because solutions were submitted after less than 1 second into the task. Additionally, 3 participants were excluded from the analysis of submission times because they never submitted a solution before the time-out. This behavior suggests that these participants might have failed to understand the submission instructions.

We first analyzed participants' ability to find the optimal solution of an instance. We define *computational performance* as a dichotomous variable that is equal to 1 if the participant obtained a value equal to the maximum value obtainable in the instance, and 0 otherwise. Mean computational performance was 83.2% (min = 0.67, max = 0.94, $SD = 0.08$). Participants spent 43.5 seconds on average on an instance (min = 27.4, max = 60.0, $SD = 8.9$). Participants were allowed to select any set of items, irrespective of the capacity constraint, which implied that they could submit candidate solutions that exceeded the capacity constraint. However, the capacity constraint was only violated in 3% of instances. Performance did not change throughout the task ($P = 0.683$, main effect of trial number on performance, GLMM; Table E.7 Model 1), nor did the time spent per instance ($P = 0.483$, main effect of trial number on time, linear mixed model (LMM); Table E.8 Model 1), suggesting that neither experience nor mental fatigue affected task performance.

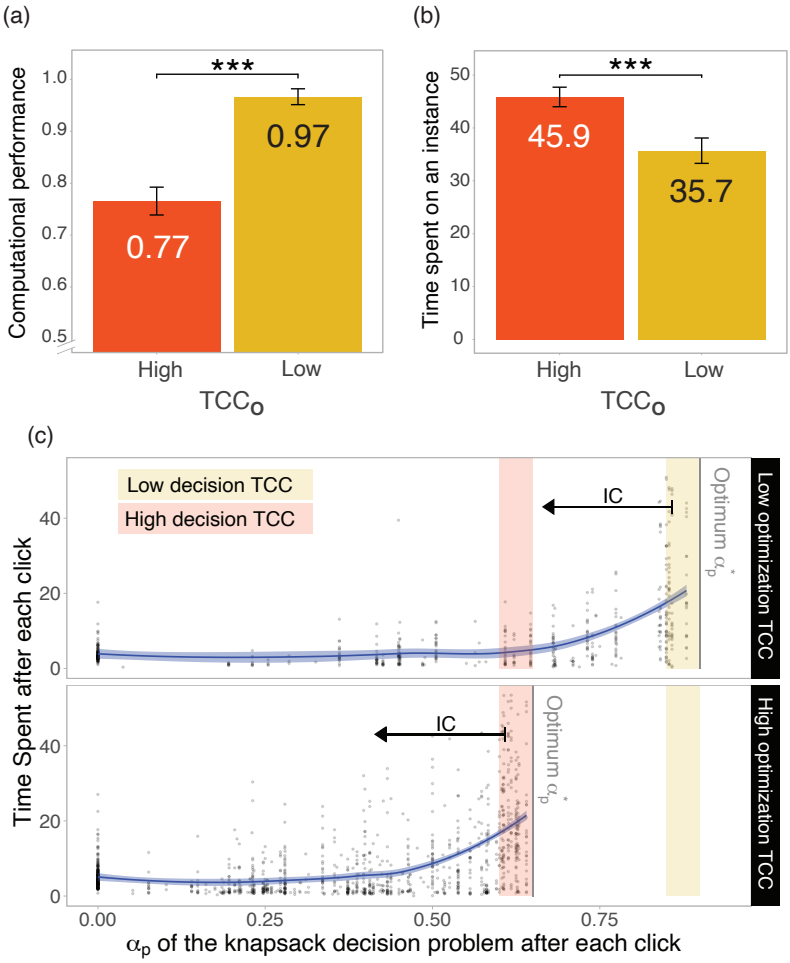
Effect of typical-case complexity on performance

We define typical-case complexity for optimization problems (TCC_O) as the TCC of the decision problem of choosing whether the optimal value is achievable (see appendix C). We hypothesized that computational performance in instances with *high* TCC_O (instances whose solutions have a corresponding decision problem with high TCC) would be lower than in instances with *low* TCC_O (instances whose solutions have a corresponding decision problem with low TCC). We found exactly this, mean computational performance was lower in instances with high TCC_O, relative to those with low TCC_O ($P < 0.001$, main effect of TCC_O, GLMM; Fig 3.4a; Table E.7 Model 2).

So far, we have defined computational performance as a dichotomous variable. We now look at a finer-grained measure. To this end, we define *item performance* as the minimum number of item replacements needed to reach the optimal solution. These include both the removal of items that are not in the optimal solution and the addition of items that are in the optimal solution (but not part of the candidate solution). The higher the value of this measure, the further away the submitted solution is from the optimum in item space. We found that item performance was worse, on average, in instances with high TCC_O, relative to instances with low TCC_O ($P < 0.001$, main effect of TCC_O, LMM; Table E.2 Model 2).

Another way of defining performance is in terms of value obtained in an instance. We define *economic performance* as the ratio of the total value of items in the

Figure 3.4: **Relation between computational complexity and human performance in the knapsack optimization task. (a) Relation between TCC_O and computational performance.** Mean computational performance and standard error of the means (SEM) in the knapsack optimization task according to TCC_O. **(b) Relation between TCC_O and time-on-task on an instance.** Mean time spent (and SEM) in the knapsack optimization task according to TCC_O. **(c) Time spent after each click and TCC.** After each click participants were faced with the question: "Is there another set of items with a higher profit that still satisfies the weight capacity constraint?". Each of these decisions is a knapsack decision problem with a corresponding α_p and corresponding TCC. The figure shows the amount of time people spent at each of these decisions before doing another click. Note that the IC of the knapsack decision problem at each click is defined as the distance between α_p and the optimum $\alpha_p^\star$. The top panel shows instances with low TCC_O; that is, optimization instances whose optimum lies in a low TCC region. The bottom panel shows optimization instances with a high TCC_O. *Note:* $^*p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$; *NS: not significant.*



submitted solution to the total value of items in the optimal solution. We found that economic performance was lower in instances with high TCC_O relative to instances with low TCC_O ($P < 0.001$, main effect of TCC_O, LMM; Table E.2 Model 1). Taken together, these results suggest that TCC_O affects performance, robustly, irregardless of the performance metric used.

Relation between performance in the knapsack decision task and the knapsack optimization task

We hypothesized that participants' performance in the two tasks would be related: participants who performed better in the knapsack decision task were expected to perform better in the knapsack optimization task. For this analysis we excluded one participant whose performance in the knapsack decision task was significantly below the performance of any other participant. We found a positive and significant correlation between performance in the two tasks (Pearson correlation = 0.67, $P = 0.002$, d.f. = 17, correlation between average performance in the decision variant and computational performance in the optimization variant). If the outlier is included, we get qualitatively similar results (Pearson correlation = 0.49, $P = 0.027$, d.f. = 18).

Relation between complexity and time-on-task

The knapsack optimization task also allowed us to investigate effort. We tracked time-on-task, since it is likely to increase in the number of computations performed and in the time required for each computation.

We hypothesized that participants would expend more time on more difficult instances. As expected, participants spent more time on instances with high TCC_O relative to those with low TCC_O ($P < 0.001$, main effect of TCC_O, LMM; Fig 3.4b; Table E.8 Model 2). This effect was also present when controlling for computational performance ($P = 0.037$, main effect of TCC_O, LMM; Table E.8 Model 4). This means that even when participants did not find the optimal solution, they expended more time on instances with high TCC_O.

Next, we analyzed the relation between time expended in an instance and performance in the instance. We found a negative relation between time-on-task and the probability of finding the solution ($P < 0.001$, main effect of time, GLMM; Table E.7 Model 5). However, when we account for TCC_O, the effect of time on performance is no longer significant ($P = 0.905$, main effect of time; $P = 0.352$, interaction effect of time and TCC_O, GLMM; Table E.7 Model 3). Taken together with previous results, it appears that the relation between time-on-task and computational performance is driven by TCC_O. As such, the negative correlation between time expended and performance may have been caused by the effect of TCC_O on time expended.

In order to further examine the relation between optimization instances, time expended and complexity, we examined the amount of time participants spent after each click at each selection of items before performing the next click. After each click participants were faced with the question: "Is there another set of items with a higher profit that still satisfies the weight capacity constraint?". Previous results in the literature suggest that, at each selection of items in the optimization task, current TCC has an effect on the time spent (Yadav et al. 2020). We found that the effect was driven by the constrainedness of the instance (Fig 3.4c): time spent after each click was mainly influenced by IC ($P < 0.001$, main effect of IC, LMM; Table E.3 Model 2) rather than by TCC ($P = 0.206$, main effect of TCC, LMM; Table E.3 Model 2). Additionally, at each click, we estimated how many subsets of items would yield a greater sum of values than the current selection, while still satisfying the capacity constraint. We found that participants spent more time

when there were fewer alternatives that yielded a more valuable solution, whilst still satisfying the capacity constraint ($P < 0.001$, main effect of the number of more valuable solutions, LMM; Table E.3 Model 1).

Problem and algorithm specific metric of hardness related to human performance

Previous work studying human performance in the knapsack optimization problem identified an algorithm-specific measure of hardness, *Sahni-k*, that correlates with human performance (Meloso, Copic, and Bossaerts 2009; Murawski and Peter Bossaerts 2016). In line with those studies, we found a negative relation between *Sahni-k* and human computational performance ($P < 0.001$, main effect of *Sahni-k*, GLMM; Table E.7 Model 4), as well as a positive relation between *Sahni-k* and time-on-task ($P = 0.001$, main effect of *Sahni-k*, LMM; Table E.8 Model 3). However, when controlling for TCC, the effect of *Sahni-k* on time-on-task is no longer significant ($P = 0.580$, main effect of *sahni-k*, LMM; Table E.8 Model 5). Our results replicate previous finding relating *Sahni-k* with human performance and time-on-task. They also suggest that the effect of *Sahni-k* is, at least partially, captured by the generic TCC metric.

3.3.3 Computational capacity and performance

Human performance stems from a tension between intrinsic computational hardness and the computational bounds of the agent. We have focused our attention thus far on the hardness of the task, and its effect on human performance, independently of the computational capacity of the agent. We study now the relation between performance in the knapsack tasks and computational capacity. In order to investigate this interaction we used tests aimed at assessing different computational capacities. Specifically, we assessed participant's mental arithmetic, working memory, episodic memory, strategy use as well as processing and psycho-motor speed. We correlated performance in these tasks with performance on the knapsack tasks. Correlations were all non-significant (see Section 3.2 and Appendix E.1 for details). These results suggest that none of the computational capacities assessed acted as a single active constraint affecting the variability in performance across participants. It is, of course, also possible that our study did not have sufficient statistical power to detect individual differences.

3.4 Discussion

Many models of decision-making, implicitly or explicitly, require the decision-maker to solve computationally intractable problems, that is, problems that are NP-hard (van Rooij et al. 2019; Peter Bossaerts and Murawski 2017). However, little is known about the generic effect of computational complexity on human decision-making. Following a popular approach to studying computational complexity of instances of NP-hard problems (Cheeseman, Kanefsky, and Taylor 1991; Krzakala et al. 2006; Selman and Kirkpatrick 1996; Achlioptas, Naor, and Peres 2005; Monasson et al. 1999; Zdeborová and Mézard 2008; Percus, Istrate, and Moore 2006; Yadav et al.

2020), we study which inherent properties of instances make instances computationally hard for people.

Specifically, we examined the effect of intrinsic complexity of instances on performance and time-on-task in the decision and optimization variants of the 0-1 knapsack problem. To this end, we derived two measures of computational hardness that are based on inherent mathematical properties of instances. The first, typical-case complexity (TCC), captures the average complexity of random ensembles of instances. The second, instance complexity (IC), is a metric of complexity of individual instances.

In both variants of the knapsack problem, we found that performance was lower in random instances with high TCC compared to instances with low TCC. Moreover, time expended was positively correlated with TCC. Instance-specific complexity (IC) had a strong effect on time-on-task and performance as well. Thus, we provide evidence that human ability to solve complex computational problems is related to mathematical properties of individual instances of such problems.

This study investigates the relation between generic structural properties of instances of a computational problem and human behavior. Our results provide strong evidence that there are inherent mathematical properties that determine hardness for both human and digital computers. This is the case despite the fact that human and digital computers are presumably based on different computational architectures. Hence, our results can be interpreted as further evidence that computational complexity is an inherent (or absolute) property of a computational problem (Arora and Barak 2009; Monasson et al. 1999; Cheeseman, Kanefsky, and Taylor 1991).

Our results have two major implications for the study of human decision-making. Firstly, they provide empirical support for the premise that computational hardness stems from intrinsic properties of the problem and thus calls for a deeper investigation of how intrinsic properties of a task should be reflected in models of cognition. Secondly, our approach provides a framework that lends itself to studying how the interaction between computational hardness and cognitive capacity affects strategy selection. In what follows, we explore each of these implications in turn.

3.4.1 Intrinsic hardness of cognitive tasks

Two different approaches stand to notice in the the study of intrinsic hardness of tasks in cognition: a primarily theoretical approach with the aim of assessing the a priori plausibility of models of cognition, and an empirical approach, like the one used in the present study, whose goal is to derive empirically testable predictions on how properties of a task affect human problem-solving capabilities, and to test those predictions.

From a computational perspective, it has been suggested that models of decision-making, and cognition more generally, ought to be computationally tractable in order to be plausible from a computational point of view (Frixione 2001; van Rooij 2008; Peter Bossaerts and Murawski 2017; Tsotsos 1990; Levesque 1988; Blum and Vempala 2020). This approach has led to the characterization of a set of human computable problems based on worst-case asymptotic computational complexity. In this context, a problem is considered intractable when the number of operations that need to be taken to find a solution grows quickly to levels that makes solving these problems infeasible. For instance, the P-Cognition Thesis proposes that computa-

tional plausibility should be linked to P-time (i.e., polynomial time) computability (Frixione 2001; Levesque 1988). However, this thesis is often considered too restrictive (Blum and Vempala 2020; van Rooij 2008). Recognizing that instances of hard problems can vary substantially in computational resource requirements, the Fixed Parameter Tractable (FPT) Cognition Thesis proposes that models of cognition may still be computable if they are P-time computable when a parameter of the problem (or a set of them) is restricted to a a small value (van Rooij 2008; van Rooij et al. 2019).

Our study both supports these proposals and complements them. It provides empirical evidence relating human decision-making capacity to individual properties of instances and shows which of those properties make individual instances hard for people. Thus, it supports the assumption of frameworks like the P-Cognition Thesis (Frixione 2001; van Rooij 2008; Peter Bossaerts and Murawski 2017; Tsotsos 1990; Levesque 1988) that computational hardness stems from intrinsic properties of the problem. Moreover, our study could provide new insights for the development of frameworks like fixed parameter tractability (van Rooij 2008; van Rooij et al. 2019). For instance, future research could explore whether TCC, beyond being a source of average-case complexity (Cheeseman, Kanefsky, and Taylor 1991; Krzakala et al. 2006; Selman and Kirkpatrick 1996; Achlioptas, Naor, and Peres 2005; Monasson et al. 1999; Zdeborová and Mézard 2008; Percus, Istrate, and Moore 2006; Yadav et al. 2020), is also a source of worst-case complexity. That is, future work could analyze whether computational problems restricted to low TCC are solvable in polynomial time (i.e., fixed-parameter tractable relative to TCC).

From an empirical perspective, the asymptotic worst-case complexity approach is too broad and not suitable for generating empirically testable predictions from metrics of intrinsic hardness of instances. This program can not be used to explain differences in performance within a problem. It categorizes problems based on the computational resources needed to solve the most difficult cases of the problem and according to their asymptotic behavior.

Many studies have explored the hardness of instances from an empirical perspective. It has been shown that task-dependent and algorithm-dependent metrics of computational complexity predict human effort and performance in cognitive tasks including decisions (Murawski and Peter Bossaerts 2016; Bourgin et al. 2017; Shepard and Metzler 1971; Dry et al. 2006; Guid and Bratko 2013; Stazyk, Ashcraft, and Hamann 1982; De Visscher and Noël 2014). This approach is problematic, though, because these metrics are not readily generalizable to other problems and we cannot usually observe which algorithms people use when solving a cognitive problem. We propose that TCC and IC are better suited to study the interaction between computational complexity and human computational bounds.

TCC is a measure that studies complexity of "typical" instances of a problem. Moreover, it captures complexity in a way that is independent of a particular algorithm or model of computation (Cheeseman, Kanefsky, and Taylor 1991; Achlioptas, Naor, and Peres 2005; Monasson et al. 1999) and it has been proven to be applicable to a large range of problems, including the graph coloring problem (Cheeseman, Kanefsky, and Taylor 1991; Krzakala et al. 2006), the traveling salesperson problem (Ian P Gent and Walsh 1996) and the K-SAT problems (Boolean satisfiability problems; Cheeseman, Kanefsky, and Taylor 1991; Achlioptas, Naor, and Peres 2005; Selman and Kirkpatrick 1996; Krzakala et al. 2006). Our findings show that TCC

has an effect on human performance, as well as on time-on-task.

We also investigated IC as an alternative metric to capture difficulty of an instance and we found a close relation between this measure and human performance. IC complements TCC by providing a measure of difficulty at an individual instance level. While TCC maps the constrainedness of an instances to the average complexity of similar instances, IC characterizes constrainedness of a single sampled instance regardless of the ensemble it came from. It is worth noting, however, that in order to compute IC, the corresponding optimization problem has to be solved, whereas TCC is a measure that can be estimated entirely based on mathematical properties of the problem. This makes TCC not only less computationally intensive, but perhaps also a better candidate for playing a role in human meta-decisions such as strategy selection (Lieder and Griffiths 2017; Lieder, Shenhav, et al. 2018).

Taken together, the results reported have an important implication for the study of human decision-making. The predictive power of current models of decision-making is inevitably limited by the large variability in strategies deployed by humans (e.g., Murawski and Peter Bossaerts 2016; Gerd. Gigerenzer and Selten 2001; Todd and Gerd Gigerenzer 2012). We postulated that performance and effort are driven, at least partially, by properties of the instance at hand rather than being a sole feature of the solver (the human), and suggesting that predictive power of decision theoretic models can be improved by including instance properties. That is, a key goal of decision-making research so far has been to identify the procedures (algorithms, heuristics) that humans deploy (A. Newell and Herbert A Simon 1972; Gerd. Gigerenzer and Selten 2001). We propose that this approach can be complemented by studying, directly, how properties of an instance affect performance and effort. There is an analogy with the study of human effort and performance in probabilistic tasks. For instance, in the restless bandit tasks inherent (stochastic) features of the task at hand predict effort and performance, and these features form the core of the "filter models" that capture the essence of human learning in bandit problems (Payzan-Lenestour and Peter Bossaerts 2011; Averbeck 2015; Daw et al. 2006).

3.4.2 Hardness and decision-making: Adaptation of strategies

In order for a theory of decision-making to be plausible from a computational perspective, the computational requirements of a decision task need to be within the cognitive resources available to a decision-maker. Indeed, it has been suggested that the principle of rationality should not (only) be applied at the level of behavior (Marr's computational level) but (also) at the level of computation (Marr's algorithmic level), an approach known as resource rationality (Lieder and Griffiths 2019). In this framework, limited computational resources are allocated to tasks in a way that is optimal relative to specified objective function (Lieder and Griffiths 2019). Our approach provides a framework that lends itself to studying which strategies are actively used. Specifically, it allows for the study of why particular heuristics or algorithms are successful on some instances but not on others and how this explains why participants' use of heuristics changes with instance properties (Lieder, Plunkett, et al. 2014; Payne, Bettman, and Johnson 1988). Moreover, our proposal presents a method to study the meta-decision of effort allocation.

The correlation between our metrics and behavior can shed light on which strategies are actually being used. The metrics of instance difficulty we employ here — TCC and IC— correlate with two aspects of the KP, namely, constrainedness and satisfiability. It is unsurprising then, one could argue, that performance and effort are related to our metrics. This is not so, as the following example shows. Imagine that humans used the greedy algorithm. In this case, a person would fill the knapsack to capacity with items that are ordered in reverse of the value-to-weight ratio. Performance is now related to satisfiability and unrelated to TCC for unsatisfiable instances. If an instance is satisfiable, the greedy algorithm may predict it is not; only if an instance is unsatisfiable will the greedy algorithm always be correct. This entails that performance in unsatisfiable instances should be 100% regardless of TCC, while satisfiable instances should have a lower accuracy. This is not what we find. It suggests that humans may actually search in a fundamentally different way. This provides further insight as to how humans approach the KP.

This approach can also explain why participants' use of heuristics changes with instance properties. For instance, certain heuristics such as the greedy algorithm may be an adequate strategy to solve instances with low constrainedness, but not for instances with high constrainedness. Thus, participants could adjust their strategy based on the level of constrainedness of an instance. It is worth noting, however, that the problem of choosing a heuristic among a set of possible heuristics can in itself be an NP-hard problem (Rich et al. 2019). Further research could explore how TCC affects strategy selection, and in particular heuristic selection.

Additionally, the meta-decision of how to allocate effort can be informed by the study of intrinsic hardness. In order to understand how limited computational resources are allocated, it is necessary to study both the cognitive capacities of decision-makers as well as the cognitive requirements of a task. Here we provide a framework to study the latter in the context of effort allocation. Empirically, evidence from the current study suggests that agents expend more time on instances with higher TCC. Theoretically, TCC is particularly suitable as an approximation of the expected computational requirements because of its characteristics. Firstly, it is an *ex-ante* measure, that is, it is based on a set of features of the task, which could potentially be identified and used by the agent *before* solving the task. Secondly, the set of features related to TCC are intrinsic to the task, that is, they are not specific to the particular algorithm used to solve a problem. Thirdly, TCC has been shown to be generalizable to a large set of computational problems (Cheeseman, Kanefsky, and Taylor 1991; Achlioptas, Naor, and Peres 2005; Selman and Kirkpatrick 1996; Yadav et al. 2020; Ian P Gent and Walsh 1996). Further research could usefully explore whether TCC is a relevant dimension in the multi-attribute decision problem of effort allocation. The meta-decision of allocating effort based on a set of inherent and generic properties of the task, such as TCC, resonates with probabilistic tasks where agents rely on inherent properties, such as mean and variance, to make a decision. However, it remains an open question whether humans compute TCC in order to estimate the expected costs of performing a task and allocate effort.

3.4.3 Directions for future research

We have studied an important aspect of difficulty, for humans, of solving random instances of the knapsack problem. Future work should explore whether our results

can be extended to other problems. Specifically, the theoretical framework of TCC has been shown to generalize to other NP-hard problems (Cheeseman, Kanefsky, and Taylor 1991; Achlioptas, Naor, and Peres 2005; Selman and Kirkpatrick 1996; Yadav et al. 2020; Ian P Gent and Walsh 1996), but it is an open question whether the applicability of TCC and IC to human problem-solving extends to these other problems as well. Moreover, the knapsack problem describes a multi-attribute decision problem with two attributes (weight and value), which is ubiquitous in daily life (Kellerer, Pferschy, and Pisinger 2004; Meloso, Copic, and Bossaerts 2009; Torralva et al. 2013). Future work could explore whether these metrics are extendable to problems with more attributes such as the multidimensional knapsack problem (Kellerer, Pferschy, and Pisinger 2004).

In our study, the optimization task involved finding the optimal solution. However, finding the exact solution might not always be required in the real-world. In many cases, finding an approximate solution might suffice. Future research should investigate whether the results found in this study can be extended to approximation.

Additionally, our results for TCC are based on a particular sampling distribution. Specifically, we used the uniform distribution to sample the knapsack instances. This approach has been used to understand hardness and to study "typical" instances of a problem, but these instances might not necessarily be the ones we encounter outside of the laboratory setting. Characterizing real-life distributions of instances is an open research question in computer science (Bogdanov and Trevisan 2006). Further research would be required to study whether this method is generalizable to other sampling distributions and, specially, to those distributions that are encountered in everyday life.

Our work provides a step towards understanding the effects of computational complexity on human behavior by providing a measure of decision difficulty based on one dimension of complexity, namely constrainedness. This dimension has been linked to *time complexity* (the number of computations needed to find a solution), but is likely not the only measure that captures difficulty of tasks for humans. There may be many other relevant cognitive dimensions, such as memory, that are relevant for understanding cognitive limitations (Otto et al. 2013; Schmeichel 2007; Blum and Vempala 2020). Further research is needed in order to incorporate the full spectrum of cognitive costs and resource limitations and link them to performance and time-on-task in decision tasks.

We have shown that TCC and IC affect behavior through task performance and time-on-task. Yet, it could also have an impact on behavior in other ways. For instance, attitudes towards complexity could affect behavior. Complexity avoidance could lead people to avoid situations that involve solving difficult tasks, whereas complexity seeking could lead to situations in which people seek tasks that require a high amount of effort to be solved (Inzlicht, Shenhav, and Olivola 2018). Another way that complexity could be related to behavior is through its effect on uncertainty. In the case of the knapsack optimization task, it is still an open question which mechanism participants used to adjust the time-on-task. TCC could influence the level of uncertainty of having found the solution, and in turn this uncertainty could play a role in the decision of when to submit an answer (Ackerman and Thompson 2017). We leave it to future work to explore the effects of attitudes towards (or preferences over) complexity in decision-making, as well as the relation between

complexity, uncertainty and behavior.

Overall, this study provides evidence that computational complexity can be characterized by inherent properties of a computational problem (Cherniak 1984). An understanding of such properties should also 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. This is crucial for the design of policies that wish to improve the quality of decisions people make and the outcomes they achieve in areas such as financial investments or the selection of health insurance contracts, among many others. In those cases where the task is too demanding, mechanisms could be designed to help people improve the quality of their decisions. This could be done, for instance, through software applications that take advantage of the computational power of electronic computers. Note, for example, that an electronic computer can correctly solve an instance of the knapsack problem (with six items) in less than one second, whereas humans in our study took 25 seconds, and they didn't always find the optimum. Finally, our results advocate for closer collaboration between decision scientists and computer scientists. Not only can decision sciences be informed by computation theory, as was done in this study, but research on humans could motivate the development of new theories and algorithms.

Acknowledgments

This research is supported by a University of Melbourne Graduate Research Scholarship from the Faculty of Business and Economics. Bossaerts acknowledges financial support through a R@MAP Chair from the University of Melbourne.

Competing interests

The authors declare no competing interests.

Appendices

Appendix A Instance sampling

A.1 Knapsack decision problem

All instances in the experiment had 6 items. The probability that a particular instance is satisfiable can be characterized in terms of the normalized capacity constraint ($\alpha_c = \frac{c}{\sum_{i=1}^{n} w_i}$) and the normalized target profit ($\alpha_p = \frac{p}{\sum_{i=1}^{n} v_i}$). Importantly, these parameters, together with the satisfiability probability, characterize TCC as well.

We made use of this property to select instances for the task, as follows. We first sampled (with replacement) a collection of 250 combinations of weights and values ($< w_1, ..., w_6 >< v_1, ..., v_6 >$) from a uniform (and discrete) distribution over the range 1 to 50. For every weight/value combination, multiple knapsack instances were

generated by increasing (in discrete steps) both capacity and the profit constraints from a lower bound of 1 to an upper bound equal up to the sum of weights ($\sum_{i=1}^{n} w_i$) for the capacity constraint (c) and the sum of values ($\sum_{i=1}^{n} w_i$) for the target profit (p). Target profits and capacities were rounded to the nearest integer and repeated instances were omitted. This process generated a total of 2,496,603 instances. Using these instances, we binned them into bins of width 0.05×0.05 according to their normalized capacity (α_c) and normalized profit (α_p). This allowed us to estimate for each one of these bins the probability that the instance was satisfiable.

From the instances generated we selected the normalized capacity (α_c) bin of $[0.40, 0.45]$ and chose the normalized profit bins that corresponded to the under-constrained (low TCC; $\alpha_p \in [0.35, 0.4]$), satisfiability threshold region (high TCC; $\alpha_p \in [0.6, 0.65]$) and over-constrained (low TCC; $\alpha_p \in [0.85, 0.9]$) regions. We then randomly selected 18 instances from the under-constrained bin and 18 from the over-constrained bin. Finally, we sampled 18 *satisfiable* instances and 18 *non-satisfiable* instances from the satisfiability threshold bin (0.4-0.45). Throughout we ensured that no weight/value combinations were sampled twice. In order to also ensure enough variability among instances in the satisfiability threshold region we added an additional constraint in the sampling from each bin. We forced half of the instances selected in each of the bins close to the satisfiability threshold (high TCC) to be easier than the median according to an algorithm specific ex-post complexity measure and the other half to be harder than the median.

For this, we made use of an ex-post complexity measures based on a generic off-the-shelf solver *Gecode* (Gecode Team 2006). Gecode is a constraint solver that uses a constraint propagation technique with different search methods, such as branch-and-bound. We implemented this solver using Minizinc (Nethercote et al. 2007). A natural *algorithm-specific* complexity metric would be the time required for the algorithm to solve the problem. However, computational time was not directly used given that instances with only 6 items are solved rapidly by a computer, and thus the signal-to-noise ratio of this measure is low. Instead, we chose complexity measures that provide a good proxy of the search effort. Explicitly, we explored how different proxies correlated with computational time when solving knapsack decision instances with 15, 20, 25 and 30 items. We found that the number of propagations had the highest correlations to computational time in the knapsack decision problem (table E.5). This metric is an approximation of the number of options available for exploration after constraint implementation. We used this measure as a proxy for ex-post complexity to ensure enough variability among instances with high TCC.

A.2 Knapsack optimization problem

To generate instances for the task, a sampling process similar to the one for the knapsack decision task was used. Taking the same combinations of weights and values sampled for the knapsack decision task, a series of optimization problem instances were generated by increasing (in discrete steps) the capacity constraint from a lower bound of 1 to an upper bound equal to the sum of weights ($\sum_{i=1}^{n} w_i$). Capacities were rounded to the nearest integer and repeated instances were omitted. This resulted in 24,892 instances. To compute the optimization typical-case complexity (TCC_O) of each of these instances, we computed the normalized value of the solution (α_p^*; see appendix C for details). We selected the same normalized capacity bin as

for the knapsack decision task ($\alpha_c \in [0.4, 0.45]$) and selected the normalized profit of the solution such that the corresponding decision problem lied in the satisfiability threshold region (high TCC_O; $\alpha_p^* \in [0.6, 0.65]$) and in the over-constrained region (low TCC_O; $\alpha_p^* \in [0.85, 0.9]$). It is worth noting that the instance generation process did not produce instances in the under-constrained region ($\alpha_p^* \in [0.35 - 0.4]$). Again, we forced half of the instances selected in each of the bins in the satisfiability threshold region (high TCC_O) to be easier than the median, according to the *Gecode* propagations measure, and the other half to be harder than the median. We sampled a total of 18 instances, 12 with high TCC_O and 6 with low TCC_O.

Even though we used again *Gecode* propagations as a proxy for computational time to ensure enough variability among instances with high TCC; it is worth noting that the Gecode solver is not entirely comparable across the two variants of the knapsack problem. Solving different problems might involve different algorithms within one solver. Therefore, we verified whether the number of propagations could be used as a proxy for the optimization variant as well and found that the correlations of the complexity measures to computational time were similar across both problems (table E.5).

Appendix B CANTAB tasks

Four tests from the Cambridge Neuropsychological Test Automated Battery (CANTAB; Cognition 2017) are used to measure certain aspects of cognitive function such as working memory and strategy use.

Reaction Time (RTI) Five yellow circles are displayed at the top of the screen, whilst the participant must press and hold down a touchscreen button at the bottom of the screen. When a spot appears inside one of the yellow circles the participant must respond as quickly as possible by letting go of the button and touching the circle where the yellow spot had appeared. This is repeated for 30 trials.

Paired Associates Learning (PAL) Boxes are displayed on the screen and open one by one in a randomized order to reveal patterns hidden inside. The patterns are then displayed in the middle of the screen, one at a time, and the subject must touch the box where the pattern was originally located.

Spatial Working Memory (SWM) The test begins with colored boxes being shown on the screen. The aim of this test is that, by touching the boxes and using a process of elimination, the subject should find one 'token' in each of the boxes and use them to fill up an empty column on the right hand side of the screen. The computer will never hide a token in the same colored box, so once a token is found in a box the participant should not return to that box to look for another token.

Spatial Span Task (SSP) White squares briefly change color in a variable sequence. The participant must remember the sequence and then touch the squares in that same order. The sequence length increases through the test. There are up to 3 attempts at each sequence length and the test terminates if all three are failed.

Appendix C Extension of TCC to the knapsack optimization problem

The characterization of complexity using typical-case complexity (TCC) theory is based on the satisfiability probability and therefore is only applicable to decision problems. However, in everyday life we are likely to encounter optimization problems. In order to generalize the TCC measure we frame the knapsack optimization problem (KOP) as a sequence of decision problems in which the question to solve can be reduced to a chain of questions: "Is there another set of items with a higher profit that still satisfies the weight capacity constraint?". In other words, we model the decision-maker as selecting a subset of items that satisfy the capacity constraint and then decides whether there exist another combination that would yield them a higher profit and still satisfy the constraint. If the answer is 'yes', the agent chooses one of such combinations and asks himself the same question again. This process is repeated until the answer is no, which means that the optimum has been reached. In order to incorporate this approach into a TCC measure for optimization problems, we generated a mapping of each KOP instance into a knapsack decision problem (KDP) instance in which the optimum value of the KOP was used as the capacity of the KDP instance (Definition 3.3). The TCC for the optimization problem (TCC_O) was defined as the TCC of the corresponding KDP. Just as with the TCC of the KDP, we expected performance of human participants to be lower in those instances that map into KDP instances with high TCC.

Definition 3.1. *0-1 Knapsack Optimization Problem (KOP) Let $n \in \mathbb{N}$, $c \in \mathbb{R}$ and $\psi = \{(w_1, v_1), ..., (w_n, v_n)\} \in \mathbb{R}^n \times \mathbb{R}^n$. The KOP is a mapping $\mathring{K}_n^{(c)}(\psi) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ such that:*

$$\mathring{K}_n^{(c)}(\psi) = \max_{\mathbf{x}} \sum_{i=1}^{n} x_i v_i$$
$$\text{subject to } \sum_{i=1}^{n} x_i w_i \leq c,\ x_i \in \{0,1\}, i = 1, ..., n.$$

Definition 3.2. *Knapsack Decision Problem (KDP)*

Let $n \in \mathbb{N}$, $c, p \in \mathbb{R}$ and $\psi = \{(w_1, v_1), ..., (w_n, v_n)\} \in \mathbb{R}^n \times \mathbb{R}^n$. A KDP is a mapping $K_n^{(c,p)} : \mathbb{R}^n \times \mathbb{R}^n \to \{0,1\}$ such that:

$$\bar{K}_n^{(c,p)}(\psi) = \begin{cases} 1 & \text{if } \exists A \subseteq \psi \text{ s.t.} \\ & \sum_{(w_i,v_i)\in A} w_i \leq c \text{ and } \sum_{(w_i,v_i)\in A} v_i \geq p \\ 0 & \text{otherwise} \end{cases}$$

Definition 3.3. *A mapping m from a KOP to a KDP. m is defined as a mapping from a KOP instance $\mathring{K}(\cdot)$ to a KDP instance $\bar{K}'_n(\cdot)$ such that:*

$$\bar{K}'^{(c,p)}_n(\psi) = \bar{K}^{(c,p')}_n(\psi) \quad \text{where} \quad p' = \mathring{K}^{(c)}_n(\psi)$$

Definition 3.4. *TCC_O of a KOP $\mathring{K}(\cdot)$ is defined as*

$$TCC_O(\mathring{K}^{(c)}_n(\psi)) = TCC(\bar{K}^{(c,p')}_n(\psi))$$

Appendix D Expected number of solution witnesses and the constrainedness of the solution space

In this section we characterize mathematically the expected *number of witnesses* (number of subsets of items that satisfy the constraints) of a random ensemble of instances of the knapsack decision problem. We start by presenting alternative definitions of the knapsack problem that will be useful to characterize the expected number of witnesses. Afterwards, we introduce the Dirichlet distribution, which will be used to model the distribution of random instances. We then characterize mathematically the expected number of witnesses. Finally, we show how the expected number of witnesses can be summarized into a single constrainedness parameter and how this is related to computational requirements.

D.1 Defining the knapsack problem

We are interested in the KDP (definition 3.2); however, to analyze the characteristics of the problem we investigate an analogous version of the problem that relates KDP to the normalized capacity ($\alpha_c = c/\sum_{i=1}^{n} w_i$) and normalized profit ($\alpha_p = p/\sum_{i=1}^{n} v_i$). We call this the normalized knapsack decision problem (NKDP), which is defined on a simplex:

Definition 3.5. *m-simplex*

$$\triangle_m = \left\{(x_1, \ldots, x_{m+1}) \in \mathbb{R}^{m+1} | \sum_{i=1}^{m} x_i = 1 \text{ and } x_i \geq 0 \ \forall i\right\}$$

Definition 3.6. *Normalized Knapsack Decision Problem (NKDP)*

Let $n \in \mathbb{N}$ and $\alpha_c, \alpha_p \in H = \{x \in \mathbb{R} | 0 \leq x \leq 1\}$. A NKDP is a mapping $K_n^{(\alpha_c,\alpha_p)} : \triangle_{n-1} \times \triangle_{n-1} \to \{0,1\}$ such that for any $\psi = \{(w_1, v_1), ..., (w_n, v_n)\} \in \triangle_{n-1} \times \triangle_{n-1}$:

$$K_n^{(\alpha_c,\alpha_p)}(\psi) = \begin{cases} 1 & \text{if } \exists A \subseteq \psi \text{ s.t. } \sum_{(w_i,v_i)\in A} w_i \leq \alpha_c \\ & \qquad \text{and } \sum_{(w_i,v_i)\in A} v_i \geq \alpha_p \\ 0 & \text{otherwise} \end{cases}$$

There is a correspondence between definitions 3.2 and 3.6; by normalizing the weights and values of the former we obtain the latter. Explicitly, if $\psi = \{(w_1, v_1), ..., (w_n, v_n)\} \in \mathbb{R}^n \times \mathbb{R}^n$ and $c, p \in \mathbb{R}$, we get that:

$$\bar{K}_n^{(c,p)}(\psi) = K_n^{(\alpha_c,\alpha_p)}(\hat{\psi})$$

where $\hat{\psi} = \{(\hat{w}_1, \hat{v}_1), ..., (\hat{w}_n, \hat{v}_n)\}$ with $\hat{w}_i = w_i/\sum_{i=1}^{n} w_i$, $\hat{v}_i = v_i/\sum_{i=1}^{n} v_i$, $\alpha_c = c/\sum_{i=1}^{n} w_i$ and $\alpha_p = p/\sum_{i=1}^{n} v_i$.

To simplify notation we now introduce an alternative version of definition 3.6:

Definition 3.7. *Normalized Knapsack Decision Problem (NKDP): Analogous Definition*

Let $n \in \mathbb{N}$ and $\alpha_c, \alpha_p \in H = \{x \in \mathbb{R} | 0 \leq x \leq 1\}$. A NKDP is a mapping $K_n^{(\alpha_c,\alpha_p)} : \triangle_{n-1} \times \triangle_{n-1} \to \{0,1\}$ such that for any $\psi = \{(w_1, v_1), ..., (w_n, v_n)\} \in \triangle_{n-1} \times \triangle_{n-1}$:

$$K_n^{(\alpha_c,\alpha_p)}(\psi) = \begin{cases} 1 & \text{if } \exists S \subseteq \{1,2,...,n\} \text{ s.t.} \\ & \quad \sum_{i \in S} w_i \leq \alpha_c \text{ and } \sum_{i \in S} v_i \geq \alpha_p \\ 0 & \text{otherwise} \end{cases}$$

Based on Definition 3.7, we define a *subset specific* NKDP. This is defined as a NKDP in which the question is whether a specific subset of items (e.g., items one and three) satisfy the normalized profit (α_p) and normalized capacity (α_c) constraints. ssNKDP is defined for every set S in the power set $\mathcal{P}(\{1,2,...,n\})$.

Definition 3.8. *Subset Specific Normalized Knapsack Decision Problem (ssNKDP)*
Let $n \in \mathbb{N}$, $\alpha_c, \alpha_p \in H = \{x \in \mathbb{R} | 0 \leq x \leq 1\}$ and $S \in \mathcal{P}(\{1,2,...,n\})$. A ssNKDP is a mapping $K_n^{(\alpha_c,\alpha_p,S)} : \triangle_{n-1} \times \triangle_{n-1} \to \{0,1\}$ such that for any $\psi = \{(w_1, v_1), ..., (w_n, v_n)\} \in \triangle_{n-1} \times \triangle_{n-1}$:

$$K_n^{(\alpha_c,\alpha_p,S)}(\psi) = \begin{cases} 1 & \text{if } \sum_{i \in S} w_i \leq \alpha_c \text{ and } \sum_{i \in S} v_i \geq \alpha_p \\ 0 & \text{otherwise} \end{cases}$$

D.2 Sampling instances: The Dirichlet distribution

We now turn our attention to the process of generation of random instances. We focus our attention on the Dirichlet probability distribution, which has been widely studied and is particularly suited to describe distributions over a simplex. This will allow us to characterize the expected number of witnesses of the NKDP in the following section. In this section we define the Dirichlet distribution and present some relevant properties.

Definition 3.9. *Dirichlet Distribution*
A random vector $(X_1, \ldots, X_m) \in \triangle_{m-1}$ is said to follow a Dirichlet distribution with parameters $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_m) \in (\mathbb{R}^+)^m$

$$(X_1, \ldots, X_m) \sim Dir(\alpha_1, \ldots, \alpha_m)$$

if the density of $(X_1, \ldots, X_{m-1})$ is

$$\frac{\Gamma(\sum_{i=1}^m \alpha_i)}{\prod_{i=1}^m \Gamma(\alpha_i)} \prod_{i=1}^m X_i^{\alpha_i - 1}$$

Lemma 3.1. *Additive Property*
Let $m \in \mathbb{N}$ and $S \subseteq \{1, 2, \ldots, m\}$. If $(X_1, \ldots, X_m) \sim Dir(\alpha_1, \ldots, \alpha_m)$ then

$$\left(\sum_{i \in S} X_i, \boldsymbol{X}_{-S} \right) \sim Dir\left(\sum_{i \in S} \alpha_i, \boldsymbol{\alpha}_{-S} \right)$$

Lemma 3.2. *Marginal Distribution*
If $(X_1, \ldots, X_m) \sim Dir(\alpha_1, \ldots, \alpha_m)$ then

$$X_k \sim Beta\left(\alpha_k, \left(\sum_{i=1}^m \alpha_i \right) - \alpha_k \right)$$

Corollary 3.1. *Let $m \in \mathbb{N}$ and $S \subseteq \{1, 2, \ldots, m\}$. If $(X_1, \ldots, X_m) \sim Dir(\alpha_1, \ldots, \alpha_m)$ then*

$$\sum_{i \in S} X_i \sim Beta\Big(\sum_{i \in S} \alpha_i, \sum_{i \notin S} \alpha_i\Big)$$

Let us turn back now to the knapsack problem and characterize the random generation process of instances. Let weights and values be independently sampled from two Dirichlet distributions:

$$(w_1, \ldots, w_n) \sim Dir(\alpha, \ldots, \alpha)$$

$$(v_1, \ldots, v_n) \sim Dir(\beta, \ldots, \beta)$$

where $\alpha, \beta \in \mathbb{R}^+$. For simplicity we will restrict ourselves to the case where the n weights and n values are both sampled uniformly from the $(n-1)$-simplex:

$$(w_1, \ldots, w_n) \sim Dir(1, \ldots, 1)$$

$$(v_1, \ldots, v_n) \sim Dir(1, \ldots, 1)$$

The uniform case can be generalized easily to any other values of α and β.

D.3 Expected number of witnesses

Our aim is to characterize the expected number of witnesses of the NKDP when we introduce randomness in the selection of the items $\psi \in \triangle_{n-1} \times \triangle_{n-1}$. In order to do this we define first the random variable that captures the number of witnesses of a ssNKDP:

Definition 3.10. *Number of Witnesses of a ssNKDP*
Let $X_n^{(\alpha_c, \alpha_p, S)} : \triangle_{n-1} \times \triangle_{n-1} \to \mathbb{N}$ such that

$$X_n^{(\alpha_c, \alpha_p, S)}(\psi) = K_n^{(\alpha_c, \alpha_p, S)}(\psi)$$

We now define the random variable of the number of witnesses of the NKDP as the sum of $X_n^{(\alpha_c, \alpha_p, S)}(\psi)$ over all possible subsets $S \in \mathcal{P}\{1, \ldots, n\}$:

Definition 3.11. *Number of Witnesses of a NKDP*
Let $X_n^{(\alpha_c, \alpha_p)} : \triangle_{n-1} \times \triangle_{n-1} \to \mathbb{N}$ such that

$$X_n^{(\alpha_c, \alpha_p)}(\psi) = \sum_{S \in \mathcal{P}(\{1, \ldots, n\})} X_n^{(\alpha_c, \alpha_p, S)}(\psi)$$

In order to calculate the expected value of $X_n^{(\alpha_c, \alpha_p)}(\psi)$ it suffices to find the expected value of $X_n^{(\alpha_c, \alpha_p, S)}(\psi)$:

$$\begin{aligned} E\Big[X_n^{(\alpha_c, \alpha_p)}(\psi)\Big] &= E\left[\sum_{S \in \mathcal{P}(\{1, \ldots, n\})} X_n^{(\alpha_c, \alpha_p, S)}(\psi)\right] \\ &= \sum_{S \in \mathcal{P}(\{1, \ldots, n\})} E\Big[X_n^{(\alpha_c, \alpha_p, S)}(\psi)\Big] \end{aligned}$$

Given that $X_n^{(\alpha_c,\alpha_p,S)}(\psi) \in \{0,1\}$ we obtain that:

$$E\Big[X_n^{(\alpha_c,\alpha_p,S)}(\psi)\Big] = 1 \times P\Big(\sum_{i\in S} w_i \leq \alpha_c \wedge \sum_{i\in S} v_i \geq \alpha_p\Big)$$

for an arbitrary $S \subseteq \{1,\ldots,n\}$ and arbitrary $\alpha_c, \alpha_p \in H = \{x \in \mathbb{R} | 0 \leq x \leq 1\}$.

Given that the weights are sampled independently, we have that

$$P\Big(\sum_{i\in S} w_i \leq \alpha_c \wedge \sum_{i\in S} v_i \geq \alpha_p\Big) = P\Big(\sum_{i\in S} w_i \leq \alpha_c\Big)\left(1 - P\Big(\sum_{i\in S} v_i \leq \alpha_p\Big)\right)$$

Additionally, by Corollary 3.1, we can conclude that

$$\sum_{i\in S} w_i \sim Beta(|S|, n-|S|)$$

$$\sum_{i\in S} v_i \sim Beta(|S|, n-|S|)$$

where $|S|$ is the cardinality of the set. Therefore, if we denote the cumulative distribution of $Y \sim Beta(p,q)$ by $I_y(p,q)$, we get that

$$P\Big(\sum_{i\in S} w_i \leq \alpha_c \wedge \sum_{i\in S} v_i \geq \alpha_p\Big) = I_{\alpha_c}\big(|S|, n-|S|\big)\big(1 - I_{\alpha_p}(|S|, n-|S|)\big).$$

Thus $E\Big[X_n^{(\alpha_c,\alpha_p)}(\psi)\Big] =$

$$\sum_{S\in\mathcal{P}(\{1,\ldots,n\})} I_{\alpha_c}\big(|S|, n-|S|\big)\big(1 - I_{\alpha_p}(|S|, n-|S|)\big)$$
$$= \sum_{j=1}^{n} \binom{n}{j} I_{\alpha_c}\big(j, n-j\big)\big(1 - I_{\alpha_p}(j, n-j)\big)$$

We summarize the above in the following result.

Result 3.1. *NKDP expected number of witnesses*

Let $n \in \mathbb{N}$ and $\alpha_c, \alpha_p \in H = \{x \in \mathbb{R} | 0 \leq x \leq 1\}$. If weights are sampled independently from values from the following distributions: $(w_i,\ldots,w_n) \sim Dir(1,\ldots,1)$ and $(v_i,\ldots,v_n) \sim Dir(1,\ldots,1)$, then the expected number of witnesses for the NKDP is given by

$$E\Big[X_n^{(\alpha_c,\alpha_p)}(\psi)\Big] = \sum_{j=1}^{n} \binom{n}{j} I_{\alpha_c}\big(j, n-j\big)\big(1 - I_{\alpha_p}(j, n-j)\big)$$

In order to relate this result to the KDP we need to find a way of mapping a distribution of weights and values in $\mathbb{R}^n$ to a distribution in $\triangle_{n-1}$ and, in particular, to the Dirichlet distribution. Fortunately, every Dirichlet distribution can be constructed from independent Gamma distributions:

Lemma 3.3. *$(X_1, \ldots, X_m) \sim Dir(\alpha_1, \ldots, \alpha_m)$ if and only if*

$$X_i \sim \frac{Y_i}{\sum_{i=1}^{m} Y_i} \text{ for } i = 1, \ldots, n$$

where $Y_i \sim Gamma(\alpha_i, 1)$ and $\{Y_i\}_{i=1}^{m}$ are mutually independent.

This implies the following result for KDP:

Result 3.2. *KDP Expected number of witnesses*

Let $n \in \mathbb{N}$ and $\alpha_c, \alpha_p \in H = \{x \in \mathbb{R} | 0 \leq x \leq 1\}$. If $w_i \sim Gamma(1,1)$ and $v_i \sim Gamma(1,1)$ are all mutually independent then

$$E\Big[\bar{X}_n^{(\alpha_c,\alpha_p)}(\psi)\Big] = \sum_{j=1}^{n} \binom{n}{j} I_{\alpha_c}\big(j, n-j\big)\big(1 - I_{\alpha_p}(j, n-j)\big)$$

where $\bar{X}_n^{(\alpha_c,\alpha_p)}(\psi)$ is defined for the KDP analogously to $X_n^{(\alpha_c,\alpha_p)}(\psi)$ for the NKDP (definition 3.11).

Using these results it is possible to calculate the expected number of witnesses for each $\alpha_c, \alpha_p \in H = \{x \in \mathbb{R} | 0 \leq x \leq 1\}$. We plot these values for different number of items (Fig D.1).

D.4 Constrainedness, satisfiability probability and computational requirements

The expected number of witnesses is tightly connected to the satisfiability probability. In particular, it has been suggested that the same parameters that characterizes the satisfiability probability (i.e α_c and α_p) characterize the expected number of witnesses. This has already been shown to apply to Boolean satisfiability problem (SAT), graph coloring and number partitioning (Ian P. Gent et al. 1996). In the previous section we showed further support for this claim by showing that, like the satisfiability probability, the expected number of witnesses of KDP (for a fixed number of items n) is characterized by the same parameters, namely the normalized profit α_p and the normalized capacity α_c. This suggests that the parameters related to the satisfiability probability in a computational problem can be found by deriving the analytical expression of the expected number of witnesses.

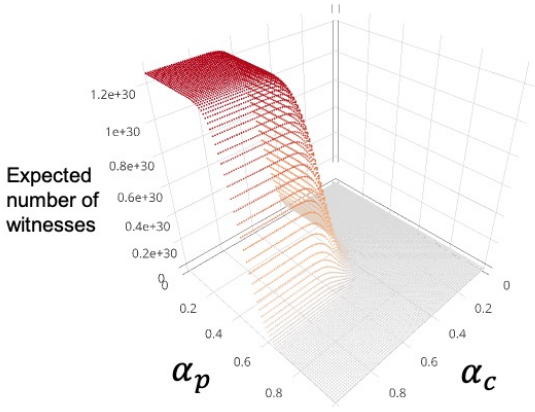
Notably, Ian P. Gent et al. 1996 proposed that the parameter κ, which depends on the expected number of witnesses, characterizes the constrainedness of search. Explicitly,

$$\kappa = 1 - \frac{log_2(\text{Expected number of witnesses})}{log_2(|states|)}$$

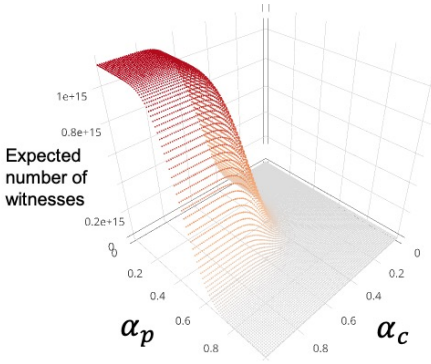
where *states* is the total state space of the problem.

We explore how the κ-parameter is related to the satisfiability phase transition and to computational requirements in the KDP. In order to do this, we first calculate κ. Note that in the knapsack, the states corresponds to all of the possible subsets of items (i.e. $S \in \mathcal{P}(\{1, \ldots, n\})$). In particular, we have that

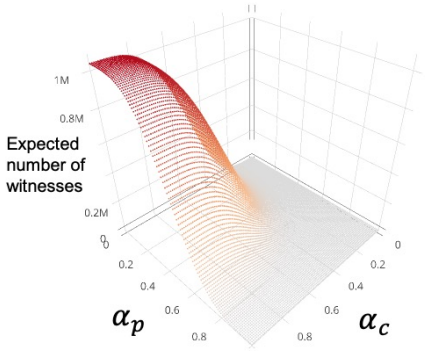
$$|states| = |\mathcal{P}(\{1, \ldots, n\})| = 2^n$$



(a) n=100



(b) n=50



(c) n=20

Figure D.1: **Expected number of witnesses in the knapsack decision problem for different number of items (n).** Weights and values are sampled from independent uniform Dirichlet distributions. Equivalently, each value and weight is sampled from an independent $Gamma(1, 1)$ distribution and then normalized with respect to the sum of values and weights, respectively.

therefore, for the knapsack decision problem

$$\begin{aligned}\kappa &= 1 - \frac{log_2\Big(E\Big[\bar{X}_n^{(\alpha_c,\alpha_p)}(\psi)\Big]\Big)}{n} \\ &= 1 - \frac{1}{n} log_2\Big[\sum_{j=1}^{n}\binom{n}{j} I_{\alpha_c}\big(j, n-j\big)\big(1 - I_{\alpha_p}(j, n-j)\big)\Big]\end{aligned}$$

To explore how κ is related to the satisfiability phase transition and the computational requirements of solving an instance we sampled 62,500 instances of the knapsack decision problem with $n = 50$ items. We solved the instances using the Gecode solver (Gecode Team 2006) and calculated the satisfiability probability across the $\alpha_c \times \alpha_p$ space. Additionally, we calculated a proxy for the average computational time required to solve instances (the number of propagations). We found that, in

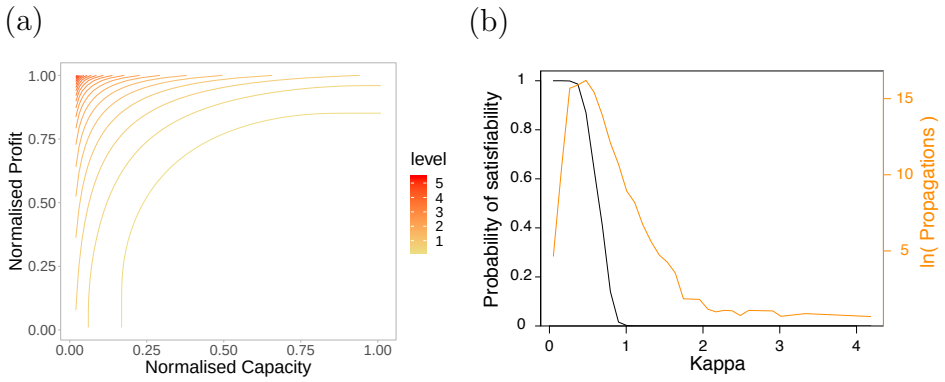


Figure D.2: κ **for the Knapsack Decision Problem with 50 items.** Weights and values were independently sampled from a $Gamma(1, 1)$ distribution. **(a) Isocurves with varying levels of κ in the $\alpha_c \times \alpha_p$ space.** Each isocurve represents values of (α_c, α_p) at which instances have the same level of constrainedness. **(b) κ, satisfiability probability and computational requirements**. Satisfiability probability (probability of the existence of at least one witness; left axis) and a proxy of solve-time (number of propagations; right axis). Both curves where estimated from empirical simulations using the Gecode solver.

line with Ian P. Gent et al. 1996, κ characterizes a phase transition in the satisfiability probability (Fig D.2b). Furthermore, we found that the average computational requirements peak around the same value of κ where the phase transition occurs (Fig D.2b). These results give support to the claim that κ characterizes a phase transition in the satisfiability probability, which is closely related to the expected computational requirements of solving an instance of the problem.

Appendix E Tables

Table E.1: **Pearson correlation between knapsack task performance and cognitive abilities.** Performance in the knapsack decision task is characterized by accuracy and in the knapsack optimization task is characterized by computational performance. The cognitive abilities measured used were mental arithmetic, episodic memory (PAL-FAMS28), working memory (SSPFSL), strategy use (SWMS) and spatial working memory (weighted SWMTE, with errors on easier tasks being weighted more). Results are shown without multiple comparisons correction.

Task	Knapsack decision	Knapsack optimization
Mental arithmetic	0.022 (0.236) p=0.926	0.359 (0.220) p=0.120
Episodic memory	-0.113 (0.234) p=0.637	0.191 (0.231) p=0.420
Working memory	-0.019 (0.236) p=0.936	0.210 (0.230) p=0.374
Strategy use	-0.342 (0.221) p=0.140	-0.351 (0.221) p=0.129
Spatial working memory	-0.165 (0.232) p=0.487	-0.325 (0.222) p=0.162
Degrees of freedom	18	18

Table E.2: **Mixed effects linear regressions on other performance measures in the knapsack optimization task.** Other measures of performance in the knapsack optimization task. Linear regressions with random intercept effects for participants relating optimization typical-case complexity (TCC_O) to economic performance (1), and item performance (2).

	Dependent variable	
	Economic performance	Item performance
	(1)	(2)
TCC_O	-0.015***	0.410***
	(0.004)	(0.092)
	$p < 0.001$	$p < 0.001$
Constant	0.999***	0.076
	(0.003)	(0.076)
Observations	347	358
Log likelihood	685.782	-441.574
Akaike Inf. Crit.	-1,363.564	891.147
Bayesian Inf. Crit.	-1,348.166	906.669
Note:	*p<0.1; **p<0.05; ***p<0.01	

Table E.3: **Effect of the number of item-subsets that perform better than the current selection of items on time before the next click.** Time spent after each item selection in the knapsack optimization task. Linear regressions with random intercept effects for participants relating the time-spent at each item selection with IC and TCC (2), and the number of item-subsets that perform better than the current selection of items. Each selection of items is associated with the number of item-subsets that satisfy the capacity constraint and have higher sum of values than the current selection (1).

	Dependent Variable	
	Time spent at each item selection	
	(1)	(2)
Number of item-subsets that perform better	-0.460***	
	(0.022)	
	p < 0.001	
IC		-17.276***
		(0.807)
		p < 0.001
TCC		-0.564
		(0.445)
		p = 0.206
Constant	13.240***	14.053***
	(0.531)	(0.629)
	p < 0.001	p < 0.001
Observations	1781	1,781
Log likelihood	-6,410.726	-6,392.535
Akaike Inf. Crit.	12,829.450	12,795.070
Bayesian Inf. Crit.	12,851.390	12,822.500
Note:	*p<0.1; **p<0.05; ***p<0.01	

Table E.4: **Model fit of alternative models relating human accuracy and instance complexity (IC) in the knapsack decision task.** Human accuracy and complexity in the knapsack decision task. R^2 and AIC model fit values of alternative models. Each of the models predicts average accuracy of an instance based on the complexity of the instance. 2 instances were identified as outliers and excluded from the analysis. The total number of observations in each model is $n = 70$.

Model	R^2	AIC
$accuracy = \beta_0 + \beta_{IC} \times IC$	0.296	-84.786
$accuracy = \beta_0 + \beta_{IC} \times IC^{0.5}$	0.433	-99.961
$accuracy = \beta_0 + \beta_{IC} \times IC^{0.1}$	0.529	-112.956
$accuracy = \beta_0 + \beta_{IC} \times IC^{0.05}$	0.537	-114.083
$accuracy = \beta_0 + \beta_{IC} \times IC^{0.01}$	0.542	-114.834
$accuracy = \beta_0 + \beta_{IC} \times ln(IC)$	0.328	-88.018
$accuracy = \beta_0 + \beta_{IC} \times log(IC)$	0.328	-88.018
$accuracy = \beta_0 + \beta_{TCC} \times TCC$	0.213	-77.053

Table E.5: **Gecode solver: algorithm-specific complexity measures in the knapsack problem.** Spearman's correlations between solver time and other solver output variables for different sizes of the knapsack problem. The correlations correspond to (a) the knapsack decision problem and (b) the knapsack optimization problem. All correlations are significant at $p < 0.001$.

(a) **Knapsack decision problem**

	Number of Items			
	15	20	25	30
propagations	**0.95**	**0.99**	**1.00**	**1.00**
nodes	0.95	0.98	0.99	1.00
failures	0.95	0.98	0.99	1.00
peak_depth	0.46	0.24	0.18	0.08

(b) **Knapsack optimization problem**

	Number of Items			
	15	20	25	30
propagations	**0.77**	**0.90**	**0.98**	**0.99**
nodes	0.76	0.90	0.98	0.99
failures	0.76	0.90	0.98	0.99
peak_depth	-0.16	-0.27	-0.35	-0.39

Table E.6: **Human performance in the knapsack decision task.** Logistic regressions with random intercept effects for participants relating the accuracy on an instance and trial number (1), typical-case complexity (TCC) (2), TCC and the satisfiability (3), Over-constrained and Under-constrained regions (4), Over-constrained and satisfiability threshold regions (5), TCC, the number of witnesses and satisfiability (6) and instance complexity (IC) (7).

	Dependent variable: human performance						
	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Trial number	0.005						
	(0.004)						
	p = 0.196						
TCC		-1.327***	-1.285***		-1.208***	-1.339***	
		(0.161)	(0.240)		(0.202)	(0.225)	
		p < 0.001	p < 0.001		p < 0.001	p < 0.001	
No. witnesses						0.139***	
						(0.041)	
						p = 0.001	
TCC:No. witnesses						0.448***	
						(0.101)	
						p < 0.001	
Satisfiability			-0.250			-1.427***	
			(0.271)			(0.243)	
			p = 0.355			p < 0.001	
TCC:Satisfiability			-0.084				
			(0.323)				
			p = 0.796				
Over-constrained				1.459***	0.250		
				(0.220)	(0.271)		
				p < 0.001	p = 0.355		
Under-constrained				1.208***			
				(0.202)			
				p < 0.001			
$IC^{0.01}$							2.829***
							(0.544)
							p < 0.001
Constant	1.516***	2.451***	2.584***	1.125***	2.333***	2.627***	-1.117**
	(0.150)	(0.167)	(0.224)	(0.130)	(0.206)	(0.220)	(0.544)
Observations	1,427	1,427	1,427	1,427	1,427	1,427	1,427
Log likelihood	-639.577	-602.372	-600.149	-601.946	-601.946	-581.201	-624.276
Akaike Inf. Crit.	1,285.153	1,210.744	1,210.299	1,211.891	1,211.891	1,174.402	1,254.552
Bayesian Inf. Crit.	1,300.943	1,226.534	1,236.615	1,232.944	1,232.944	1,205.982	1,270.342
Note:							*p<0.1; **p<0.05; ***p<0.01

Table E.7: **Computational performance in the knapsack optimization task.** Logistic regressions with random intercept effects for participants relating computational performance on an instance and trial number (1), optimization typical-case complexity (TCC_O) (2), TCC_O and the time spent (3), Sahni-k (4), and time spent (5).

	Dependent variable				
	Computational performance				
	(1)	(2)	(3)	(4)	(5)
Trial number	0.012				
	(0.030)				
	p = 0.683				
Time spent (scaled)			-0.089		
			(0.735)		
			p = 0.905		
Typical-case complexity (TCC_O)		-2.175***	-2.420***		
		(0.531)	(0.817)		
		p < 0.001	p = 0.004		
Time spent (scaled): TCC_O			-0.709		
			(0.760)		
			p = 0.352		
Sahni-k				-1.333***	
				(0.193)	
				p < 0.001	
Time spent					-0.072***
					(0.017)
					p < 0.001
Constant	1.512***	3.359***	3.914***	2.310***	4.975***
	(0.261)	(0.509)	(0.796)	(0.219)	(0.865)
Observations	358	358	305	358	305
Log likelihood	-161.752	-147.622	-115.185	-135.816	-124.754
Akaike Inf. Crit.	329.504	301.245	240.369	277.633	255.509
Bayesian Inf. Crit.	341.146	312.887	258.971	289.274	266.670
Note:					*p<0.1; **p<0.05; ***p<0.01

Table E.8: **Effort in the knapsack optimization task.** Linear regressions with random intercept effects for participants relating time spent on an instance and trial number (1), optimization typical-case complexity (TCC_O) (2), Sahni-k (3), TCC_O together with computational performance (4), and TCC_O together with Sahni-k (5).

	Dependent variable				
	Time spent				
	(1)	(2)	(3)	(4)	(5)
Trial number	0.095				
	(0.135)				
	p = 0.483				
TCC_O		10.114***		15.161**	9.535***
		(1.236)		(7.261)	(1.366)
		p < 0.001		p = 0.037	p < 0.001
Sahni-k			3.125***		1.489
			(0.950)		(2.687)
			p = 0.001		p = 0.580
Sahni-k:TCC_O					0.502
					(2.844)
					p = 0.861
Computational performance				-0.489	
				(7.210)	
				p = 0.946	
Computational performance : TCC_O				-6.784	
				(7.374)	
				p = 0.358	
Constant	41.802***	35.749***	41.469***	36.227***	35.498***
	(2.210)	(2.132)	(1.990)	(7.379)	(2.178)
Observations	305	305	305	305	305
Log likelihood	-1,188.894	-1,156.717	-1,181.856	-1,142.942	-1,151.532
Akaike Inf. Crit.	2,385.788	2,321.433	2,371.712	2,297.885	2,315.065
Bayesian Inf. Crit.	2,400.670	2,336.314	2,386.593	2,320.207	2,337.386
Note:					*p<0.1; **p<0.05; ***p<0.01

References

Achlioptas, Dimitris, Amin Coja-Oghlan, and Federico Ricci-Tersenghi (May 2011). "On the solution-space geometry of random constraint satisfaction problems". In: *Random Structures & Algorithms* 38.3, pp. 251–268. ISSN: 10429832. DOI: 10.1002/rsa.20323. URL: http://doi.wiley.com/10.1002/rsa.20323.

Achlioptas, Dimitris, Assaf Naor, and Yuval Peres (2005). "Rigorous Location of Phase Transitions in Hard Optimization Problems". In: *Nature* 435.7043, pp. 759–64. ISSN: 1476-4687. DOI: 10.1038/nature03602. URL: http://www.ncbi.nlm.nih.gov/pubmed/15944693.

Ackerman, Rakefet and Valerie A Thompson (2017). "Meta-Reasoning: Monitoring and Control of Thinking and Reasoning". In: *Trends in Cognitive Sciences* 21.8. DOI: 10.1016/j.tics.2017.05.004. URL: http://dx.doi.org/10.1016/j.tics.2017.05.004607.

Acuña, Daniel E. and Víctor Parada (July 2010). "People efficiently explore the solution space of the computationally intractable traveling salesman problem to find near-optimal tours". In: *PLoS ONE* 5.7. Ed. by Edward Vul, e11685. ISSN: 19326203. DOI: 10.1371/journal.pone.0011685. URL: https://dx.plos.org/10.1371/journal.pone.0011685.

Arora, Sanjeev. and Boaz. Barak (2009). *Computational complexity : a modern approach*. Cambridge University Press, p. 579. ISBN: 0521424267.

Averbeck, Bruno B. (Mar. 2015). "Theory of Choice in Bandit, Information Sampling and Foraging Tasks". In: *PLoS Computational Biology* 11.3. Ed. by Paul Schrater, e1004164. ISSN: 15537358. DOI: 10.1371/journal.pcbi.1004164. URL: https://dx.plos.org/10.1371/journal.pcbi.1004164.

Bates, Douglas et al. (2015). "Fitting Linear Mixed-Effects Models Using {lme4}". In: *Journal of Statistical Software* 67.1, pp. 1–48. DOI: 10.18637/jss.v067.i01.

Blakey, Ed (2011). "Computational complexity in non-Turing models of computation: The what, the why and the how". In: *Electronic Notes in Theoretical Computer Science* 270.1, pp. 17–28. ISSN: 15710661. DOI: 10.1016/j.entcs.2011.01.003.

Blum, Manuel and Santosh Vempala (2020). "The complexity of human computation via a concrete model with an application to passwords". In: *Proceedings of the National Academy of Sciences of the United States of America* 117.17, pp. 9208–9215. ISSN: 10916490. DOI: 10.1073/pnas.1801839117. URL: www.pnas.org/cgi/doi/10.1073/pnas.1801839117.

Bogdanov, Andrej and Luca Trevisan (2006). "Average-Case Complexity". In: *arXiv preprint cs/0606037*. URL: http://arxiv.org/abs/cs/0606037.

Bossaerts, Peter and Carsten Murawski (2017). "Computational Complexity and Human Decision-Making". In: *Trends in Cognitive Sciences* 21.12, pp. 917–929. ISSN: 1879307X. DOI: 10.1016/j.tics.2017.09.005.

Bourgin, David et al. (2017). "The Structure of Goal Systems Predicts Human Performance". In: *Proceedigns of the 39th Annual Meeting of the Cognitive Science Society*. Ed. by G Gunzelmann et al. Austin, TX: Cognitive Science Society, pp. 1660–1665.

Budzynski, Louise, Federico Ricci-Tersenghi, and Guilhem Semerjian (Feb. 2019). "Biased landscapes for random Constraint Satisfaction Problems". In: *Journal of Statistical Mechanics: Theory and Experiment* 2019.2, p. 023302.

Cappelletti, Marinella, Brian Butterworth, and Michael Kopelman (2001). "Spared numerical abilities in a case of semantic dementia". In: *Neuropsychologia* 39, pp. 1224–1239. URL: www.elsevier.com/locate/neuropsychologia.

Carruthers, Sarah, Michael E J Masson, and Ulrike Stege (2012). "Human Performance on Hard Non-Euclidean Graph Problems: Vertex Cover". In: *The Journal of Problem Solving* 5.1, p. 34. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1142.

Cheeseman, Peter, Bob Kanefsky, and William M Taylor (1991). "Where the Really Hard Problems Are". In: *The 12nd International Joint Conference on Artificial Intelligence*, pp. 331–337. ISBN: 1-55860-160-0. DOI: 10.1.1.97.3555.

Cherniak, Christopher (Dec. 1984). "Computational Complexity and the Universal Acceptance of Logic". In: *The Journal of Philosophy* 81.12, p. 739. ISSN: 0022362X. DOI: 10.2307/2026030. URL: https://www.jstor.org/stable/2026030.

Cognition, Cambridge (2017). *CANTAB® [Cognitive assessment software]*. URL: www.cantab.com.

Daw, N D et al. (2006). "Cortical substrates for exploratory decisions in humans." In: *Nature* 441.7095, pp. 876–9. ISSN: 1476-4687. DOI: 10.1038/nature04766. URL: http://www.ncbi.nlm.nih.gov/pubmed/16778890.

De Visscher, Alice and Marie Pascale Noël (2014). "The detrimental effect of interference in multiplication facts storing: Typical development and individual differences". In: *Journal of Experimental Psychology: General* 143.6, pp. 2380–2400. ISSN: 00963445. DOI: 10.1037/xge0000029.

Drugowitsch, Jan et al. (Mar. 2012). "The cost of accumulating evidence in perceptual decision making". In: *Journal of Neuroscience* 32.11, pp. 3612–3628. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.4010-11.2012. URL: https://www.jneurosci.org/content/32/11/3612.

Dry, Matthew et al. (2006). "Human Performance on Visually Presented Travelling Salesperson Problems with Varying Numbers of Nodes". In: *The Journal of Problem Solving* 1.1. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1004. URL: http://dx.doi.org/10.7771/1932-6246.1004.

Frixione, Marcello (2001). "Tractable competence". In: *Minds and Machines* 11.3, pp. 379–397. ISSN: 09246495. DOI: 10.1023/A:1017503201702.

Gecode Team (2006). *Gecode: Generic Constraint Development Environment*. URL: http://www.gecode.org.

Gent, Ian P and Toby Walsh (1996). "The TSP phase transition". In: *Artificial Intelligence* 88.1-2, pp. 349–358. ISSN: 00043702. DOI: 10.1016/S0004-3702(96)00030-6.

Gent, Ian P. et al. (1996). "The constrainedness of search". In: *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 1*. Portland. Oregon, pp. 246–252.

Gigerenzer, Gerd and Henry Brighton (2009). "Homo Heuristicus: Why Biased Minds Make Better Inferences". In: *Topics in Cognitive Science* 1.1, pp. 107–143. ISSN: 1756-8765. DOI: 10.1111/j.1756-8765.2008.01006.x.

Gigerenzer, Gerd and Wolfgang Gaissmaier (Jan. 2011). "Heuristic decision making". In: *Annual Review of Psychology* 62, pp. 451–482. ISSN: 00664308. DOI: 10.1146/annurev-psych-120709-145346.

Gigerenzer, Gerd. and Reinhard. Selten (2001). *Bounded rationality : the adaptive toolbox*. MIT Press, p. 377. ISBN: 9780262072144.

Guid, Matej and Ivan Bratko (2013). "Search-Based Estimation of Problem Difficulty for Humans". In: *Artificial Intelligence in Education. AIED 2013. Lecture Notes in Computer Science*. Ed. by Lane H.C. et al. Vol. 7926. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-39112-5{_}131.

Hanks, Timothy D. and Christopher Summerfield (Jan. 2017). *Perceptual Decision Making in Rodents, Monkeys, and Humans*. DOI: 10.1016/j.neuron.2016.12.003.

Hirtle, Stephen C. and Tommy Gärling (May 1992). "Heuristic rules for sequential spatial decisions". In: *Geoforum* 23.2, pp. 227–238. ISSN: 00167185. DOI: 10.1016/0016-7185(92)90019-Z.

Inzlicht, Michael, Amitai Shenhav, and Christopher Y Olivola (Apr. 2018). "The Effort Paradox: Effort Is Both Costly and Valued." In: *Trends in cognitive sciences* 22.4, pp. 337–349. ISSN: 1879-307X. DOI: 10.1016/j.tics.2018.01.007.

Kahneman, Daniel and Amos Tversky (1979). "Prospect Theory: An Analysis of Decision under Risk". In: *Econometrica* 47.2, pp. 263–292. ISSN: 00129682. DOI: 10.2307/1914185.

Kellerer, Hans, Ulrich Pferschy, and David Pisinger (2004). *Knapsack Problems*. Berlin, Heidelberg: Springer Berlin Heidelberg, p. 546. ISBN: 978-3-642-07311-3. DOI: 10.1007/978-3-540-24777-7.

Kotovsky, K., J. R. Hayes, and H. A. Simon (Apr. 1985). "Why are some problems hard? Evidence from Tower of Hanoi". In: *Cognitive Psychology* 17.2, pp. 248–294. ISSN: 00100285. DOI: 10.1016/0010-0285(85)90009-X.

Krzakala, Florent et al. (June 2006). "Gibbs States and the Set of Solutions of Random Constraint Satisfaction Problems". In: *Proceedings of the National Academy of Sciences of the United States of America* 104.25, pp. 10318–23. ISSN: 0027-8424. DOI: 10.1073/pnas.0703685104.

Levesque, Hector J. (1988). "Logic and the complexity of reasoning". In: *Journal of Philosophical Logic* 17.4, pp. 355–389. ISSN: 00223611. DOI: 10.1007/BF00297511.

Lieder, Falk and Thomas L. Griffiths (Nov. 2017). "Strategy selection as rational metareasoning." In: *Psychological Review* 124.6, pp. 762–794. ISSN: 1939-1471. DOI: 10.1037/rev0000075.

Lieder, Falk and Thomas L. Griffiths (2019). "Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources". In: *Behavioral and Brain Sciences* 43. ISSN: 14691825. DOI: 10.1017/S0140525X1900061X.

Lieder, Falk, Dillon Plunkett, et al. (2014). "Algorithm selection by rational metareasoning as a model of human strategy selection". In: *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*. MIT Press, pp. 2870–2878.

Lieder, Falk, Amitai Shenhav, et al. (2018). "Rational metareasoning and the plasticity of cognitive control". In: *PLoS Computational Biology* 14.4. ISSN: 15537358. DOI: 10.1371/journal.pcbi.1006043. URL: https://doi.org/10.1371/journal.pcbi.1006043.

MacGregor, James N. and Yun Chu (2011). "Human Performance on the Traveling Salesman and Related Problems: A Review". In: *The Journal of Problem Solving* 3.2, p. 1. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1090. URL: http://dx.doi.org/10.7771/1932-6246.1090.

Marino, Raffaele, Giorgio Parisi, and Federico Ricci-Tersenghi (Oct. 2016). "The backtracking survey propagation algorithm for solving random K-SAT problems". In: *Nature Communications* 7.1, pp. 1–8. ISSN: 20411723. DOI: 10.1038/ncomms12996.

Meloso, D, J Copic, and P Bossaerts (2009). "Promoting Intellectual Discovery: Patents Versus Markets". In: *Science* 323.5919, pp. 1335–1339. ISSN: 1095-9203. DOI: 10.1126/science.1158624.

Monasson, Remi et al. (1999). "Determining computational complexity from characteristic 'phase transitions'". In: *Nature* 400.6740, pp. 133–137. ISSN: 0028-0836. DOI: 10.1038/22055.

Moore, Cristopher and Stephan Mertens (2011). *The Nature of Computation*. 1st ed. Oxford University Press, p. 1004. ISBN: 9780199233212. DOI: 10.1093/acprof:oso/9780199233212.001.0001.

Murawski, Carsten and Peter Bossaerts (2016). "How Humans Solve Complex Problems: The Case of the Knapsack Problem". In: *Nature (Scientific Reports)* 6.34851. ISSN: 2045-2322. DOI: 10.1038/srep34851.

Nethercote, Nicholas et al. (2007). "MiniZinc: Towards A Standard CP Modelling Language". In: *Proceedings of the 13th International Conference on Principles and Practice of Constraint Programming*, pp. 529–543.

Newell, Allen and Herbert A Simon (1972). *Human problem solving.* Oxford, England: Prentice-Hall, pp. xiv, 920–xiv, 920.

Newell, Ben R., Nicola J. Weston, and David R. Shanks (May 2003). "Empirical tests of a fast-and-frugal heuristic: Not everyone "takes-the-best"". In: *Organizational Behavior and Human Decision Processes* 91.1, pp. 82–96. ISSN: 07495978. DOI: 10.1016/S0749-5978(02)00525-3.

Ohlsson, Stellan (2012). "The Problems with Problem Solving: Reflections on the Rise, Current Status, and Possible Future of a Cognitive Research Paradigm 1". In: *The Journal of Problem Solving* 5.1. DOI: 10.7771/1932-6246.1144. URL: http://dx.doi.org/10.7771/1932-6246.1144.

Otto, A. R. et al. (2013). "Working-memory capacity protects model-based learning from stress". In: *Proceedings of the National Academy of Sciences* 110.52, pp. 20941–20946. ISSN: 0027-8424. DOI: 10.1073/pnas.1312011110. URL: http://www.pnas.org/cgi/doi/10.1073/pnas.1312011110.

Payne, John W., James R. Bettman, and Eric J. Johnson (1988). "Adaptive Strategy Selection in Decision Making". In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* 14.3, p. 534. ISSN: 02787393. DOI: 10.1037/0278-7393.14.3.534.

Payne, John W., James R. Bettman, and Eric J. Johnson (1993). *The Adaptive Decision Maker*. DOI: 10.1017/cbo9781139173933.

Payzan-Lenestour, Elise and Peter Bossaerts (Jan. 2011). "Risk, unexpected uncertainty, and estimation uncertainty: Bayesian learning in unstable settings". In: *PLoS Computational Biology* 7.1, p. 1001048. ISSN: 1553734X. DOI: 10.1371/journal.pcbi.1001048. URL: www.ploscompbiol.org.

Percus, Allon, Gabriel Istrate, and Cristopher Moore (2006). *Computational Complexity and Statistical Physics*. Oxford University Press, p. 384. ISBN: 9780199760565.

Pudlák, Pavel (2013). *Logical foundations of mathematics and computational complexity: a gentle introduction*. 1st ed. Springer International Publishing, p. 695. ISBN: 9783319001180. DOI: 10.1007/978-3-319-00119-7{_}1.

Ricci-Tersenghi, Federico (2010). "Mathematics: Being glassy without being hard to solve". In: *Science* 330.6011, pp. 1639–1640. ISSN: 00368075. DOI: 10.1126/science.1189804.

Ricci-Tersenghi, Federico, Guilhem Semerjian, and Lenka Zdeborová (2019). "Typology of phase transitions in Bayesian inference problems". In: *Physical Review E* 99.4. ISSN: 24700053. DOI: 10.1103/PhysRevE.99.042109.

Rich, Patricia et al. (2019). "Naturalism, tractability and the adaptive toolbox". In: *Synthese*, pp. 1–36. DOI: 10.1007/s11229-019-02431-2.

Sahni, Sartaj and Sartaj (Jan. 1975). "Approximate Algorithms for the 0/1 Knapsack Problem". In: *Journal of the ACM* 22.1, pp. 115–124. ISSN: 00045411. DOI: 10.1145/321864.321873.

Schmeichel, Brandon J. (2007). "Attention control, memory updating, and emotion regulation temporarily reduce the capacity for executive control". In: *Journal of Experimental Psychology: General* 136.2, pp. 241–255. ISSN: 00963445. DOI: 10.1037/0096-3445.136.2.241.

Selman, Bart and Scott Kirkpatrick (Mar. 1996). "Critical behavior in the computational cost of satisfiability testing". In: *Artificial Intelligence* 81.1-2, pp. 273–295. ISSN: 0004-3702. DOI: 10.1016/0004-3702(95)00056-9.

Shepard, Roger N. and Jacqueline Metzler (Feb. 1971). "Mental rotation of three-dimensional objects". In: *Science* 171.3972, pp. 701–703. ISSN: 00368075. DOI: 10.1126/science.171.3972.701.

Simon, Herbert A (1956). "Rational choice and the structure of the environment". In: *Psychological Review* 63.2, pp. 129–138. ISSN: 0033295X. DOI: 10.1037/h0042769.

Simon, Herbert A (1990). "Invariants of human behavior". In: *Annual Review of Psychology* 41.1, pp. 1–19. ISSN: 00664308. DOI: 10.1146/annurev.psych.41.1.1. URL: www.annualreviews.org.

Stazyk, Edmund H., Mark H. Ashcraft, and Mary S. Hamann (1982). "A network approach to mental multiplication". In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* 8.4, pp. 320–335. ISSN: 02787393. DOI: 10.1037/0278-7393.8.4.320.

Todd, Peter M and Gerd Gigerenzer (2012). *Ecological rationality: Intelligence in the world.* Evolution and cognition. Todd, Peter M.: Cognitive Science Program, Indiana University, 1101 E. 10th St., Bloomington, IN, US, 47405, peter.m.todd@gmail.com: Oxford University Press, pp. xviii, 590–xviii, 590. ISBN: 978-0-19-531544-8 (Hardcover). DOI: 10.1093/acprof:oso/9780195315448.001.0001.

Torralva, Teresa et al. (Jan. 2013). ""Ecological" and Highly Demanding Executive Tasks Detect Real-Life Deficits in High-Functioning Adult ADHD Patients". In: *Journal of Attention Disorders* 17.1, pp. 11–19. ISSN: 1087-0547. DOI: 10.1177/1087054710389988.

Tsotsos, John K. (1990). "Analyzing vision at the complexity level". In: *Behavioral and Brain Sciences* 13.3, pp. 423–445. ISSN: 14691825. DOI: 10.1017/S0140525X00079577.

Tversky, Amos and Daniel Kahneman (Oct. 1992). "Advances in prospect theory: Cumulative representation of uncertainty". In: *Journal of Risk and Uncertainty* 5.4, pp. 297–323. ISSN: 0895-5646. DOI: 10.1007/BF00122574. URL: http://link.springer.com/10.1007/BF00122574.

Van Opheusden, Bas and Wei Ji Ma (2019). *Tasks for aligning human and machine planning*. DOI: 10.1016/j.cobeha.2019.07.002. URL: https://doi.org/10.1016/j.cobeha.2019.07.002.

Van Rooij, Iris (2008). "The Tractable Cognition Thesis". In: *Cognitive Science: A Multidisciplinary Journal* 32.6, pp. 939–984. ISSN: 0364-0213. DOI: 10.1080/03640210801897856. URL: http://doi.wiley.com/10.1080/03640210801897856.

Van Rooij, Iris et al. (Apr. 2019). *Cognition and Intractability*. Cambridge University Press. DOI: 10.1017/9781107358331.

Von Neumann, John and Oskar Morgenstern (1947). *Theory of games and economic behavior, 2nd rev. ed.* Princeton, NJ, US: Princeton University Press, pp. xviii, 641–xviii, 641.

Yadav, Nitin et al. (2020). "Is Hardness Inherent In Computational Problems? Performance Of Human And Digital Computers On Random Instances Of The 0-1 Knapsack Problem". In: *24th European Conference on Artificial Intelligence (ECAI 2020)*.

Zdeborová, Lenka and Marc Mézard (Dec. 2008). "Constraint satisfaction problems with isolated solutions are hard". In: *Journal of Statistical Mechanics: Theory and Experiment* 2008.12, P12004. ISSN: 1742-5468. DOI: 10.1088/1742-5468/2008/12/P12004.

Chapter 4

Generality of the Hardness Metrics

In this chapter I present the co-authored paper titled *"Task-independent metrics of computational hardness predict performance of human problem-solving"*. There we explore whether the metrics of computational hardness studied in the previous chapter are indeed generic metrics of hardness that affect human behavior across tasks. Specifically, we extend and compare the results found for the knapsack problem to two other canonical computational problems: the traveling salesperson problem and the Boolean satisfiability problem (3SAT).

Task-independent metrics of computational hardness predict performance of human problem-solving

Juan Pablo Franco, Karlo Doroc, Nitin Yadav, Peter Bossaerts, Carsten Murawski

Abstract

The survival of human organisms depends on our ability to solve complex tasks, which is bounded by our limited cognitive capacities. However, little is known about the factors that drive complexity of the tasks humans face and their effect on human decision-making. Here, using insights from computational complexity theory, we quantify computational hardness using a set of task-independent metrics related to the computational requirements of individual instances of a task. We then examine the relation between those metrics and human behavior and find that these metrics predict both performance and effort allocation in three canonical cognitive tasks in a similar way. Our findings demonstrate that the ability to solve complex tasks can be predicted from generic metrics of their inherent computational hardness.

4.1 Introduction

The adaptiveness of human organisms is bounded by their limited cognitive capacities, sometimes referred to as bounded rationality (Herbert A Simon 1990). In the words of Herbert Simon, "[h]uman rational behavior [...] is shaped by a scissors whose two blades are the structure of task environments and the computational capabilities of the actor" (Herbert A Simon 1990).

In the past several decades, the study of human behavior has focused predominantly on the latter. This work characterizes cognitive capacities and cognitive strategies, algorithms and heuristics people use in different task environments. It includes approaches such as the heuristics and biases program (Tversky and Kahneman 1974), resource and computational rationality (Griffiths, Lieder, and Goodman 2015; Gershman, Horvitz, and Tenenbaum 2015) as well as ecological rationality (Todd and Gerd Gigerenzer 2012), among others. However, very little is known about how properties of the task environment affects computational requirements and how they compare to bounds of human cognitive capacities.

Several studies have explored how the task environment affects the performance of specific algorithms or cognitive strategies (Murawski and Bossaerts 2016; Dry et al. 2006; Guid and Bratko 2013). However, this approach ignores the diversity in strategies implemented not only across humans, but also across situations. Even if the task environment is the same, different people might approach a task using different procedures and might change their procedures depending on the situation or their level of experience (MacGregor and Chu 2011; Hirtle and Gärling 1992; Murawski and Bossaerts 2016; Ohlsson 2012; Gerd Gigerenzer and Gaissmaier 2011; Payne, Bettman, and E. J. Johnson 1993). Understanding the interaction between task environment and an agent's computational capabilities is then particularly difficult given the lack of a generic cognitive strategy.

A principled and generic way to characterize the computational requirements of a task environment is to formalize the task as a computational problem and analyze its *problem complexity*. Here, computational requirements are typically analyzed at the level of problems, such as sorting an array of numbers. These requirements are typically expressed in terms of asymptotic worst-case growth of a resource such as compute time or memory. This means that resource requirements are characterized in terms of their growth as a function of the input size of the problem, for example, the length of the array to be sorted. Importantly, this is generally done by considering the growth in requirements in the worst-case as the input size increases. Problems with similar resource requirements, thus defined, are then grouped into complexity classes (Arora and Barak 2009).

As it stands, problem complexity is not amenable to modeling human behavior directly. Critically, although this approach can shed light on the a priori plausibility of models of human behavior (van Rooij et al. 2019), it is inadequate for the derivation of empirically testable predictions at finer detail. First, while complexity classes are based on asymptotic growth of resources, in practice many instances (that is, cases of a problem) people face are small in size (Blum and Vempala 2020). Second, complexity classes are typically based on worst-case growth of resources. This means that hardness is defined in terms of resources required to solve the most difficult instance of a problem. However, in most cases, there is substantial variation in resource requirements of instances of the same input size (Gent and Walsh 1996;

Cheeseman, Kanefsky, and Taylor 1991), and the worst case is often far away from typical, or average, cases and may not be encountered in the natural environment (Bogdanov and Trevisan 2006). Third, the approach classifies problems according to hardness, like a taxonomy, but it does not identify the sources of hardness, for example, which properties of instances make some harder than others. What would be desirable is a set of generic properties of individual instances of a class of problems that are associated with computational hardness in a way that is independent of the problem it belongs to. Similar to how properties like mean, variance and other statistics characterize the level of uncertainty in the class of probabilistic problems. However, there is as yet no analog for characterizing computational hardness.

There is limited research on how properties of instances of problems affect human problem-solving. To date, most studies are based on a problem-specific approach (MacGregor and Chu 2011; Hirtle and Gärling 1992; Kotovsky, Hayes, and H. A. Simon 1985; Shepard and Metzler 1971). Hence, their findings may not generalize to other problems. Recent theoretical advances in computer science and statistical physics provide a framework, referred to as typical-case complexity (TCC), that addresses this issue. It allows the characterization of computational hardness of individual instances of a problem. More specifically, it is concerned with the average computational hardness of random instances of a computational problem, linking structural properties of those instances to their computational complexity, independent of a particular computational model (Gent and Walsh 1996; Cheeseman, Kanefsky, and Taylor 1991; Percus, Istrate, and Moore 2006; Monasson et al. 1999; Mézard, Parisi, and Zecchina 2002). This work has identified computational 'phase transitions', which resemble phase transitions in statistical physics and which are related to computational hardness of instances. Such phase transitions have been found in a number of canonical NP-complete problems (i.e., problems that are both in NP and NP-hard) (Gent and Walsh 1996; Cheeseman, Kanefsky, and Taylor 1991; Percus, Istrate, and Moore 2006; Monasson et al. 1999; Mézard, Parisi, and Zecchina 2002), including the graph coloring problem (Cheeseman, Kanefsky, and Taylor 1991; Krzakala et al. 2006), the traveling salesperson problem (Gent and Walsh 1996) and the K-SAT problems (Boolean satisfiability problems) (Cheeseman, Kanefsky, and Taylor 1991; Selman and Kirkpatrick 1996; Krzakala et al. 2006), among others. This program has led to a deeper understanding of computational hardness by relating it to structural properties of instances. Importantly, it has identified that hardness of an instance is related to a generic instance property, namely *constrainedness* (see Fig 4.2.1). The framework has also been useful for understanding patterns in the performance of algorithms (Zdeborová and Marc Mézard 2008; Krzakala et al. 2006), and subsequently, generating more efficient algorithms (Mézard, Parisi, and Zecchina 2002).

A recent study applied this framework to study human behavior in the knapsack problem, a (NP-hard) combinatorial optimization problem (Franco et al. 2020). The study found that both effort (time-on-task) and ability to solve an instance were related to computational phase transitions, with patterns similar to those exhibited by generic constrained optimization algorithms. An important question is whether these findings generalize to other problems. If they do, then these properties related to computational complexity would be prime candidates for generic measures of computational hardness of human cognition, in the way that statistics like mean, variance and kurtosis serve as generic measures of probabilistic uncertainty in a task

(Preuschoff, Bossaerts, and Quartz 2006).

Here, using a behavioral experiment, we study the relation between a set of problem-independent measures of instance complexity and human performance in two canonical NP-complete computational problems, the Boolean satisfiability problem (3SAT) and the traveling salesperson problem (TSP). We then compare those results to results previously obtained for the 0-1 knapsack decision problem (KP) (Franco et al. 2020), to test their generalizability across NP-complete problems.

4.2 Results

Each participant solved one of the three problems: either 72 instances of the TSP task, 64 instances of the 3SAT or 72 instances of the KP (Figs 4.2.1,4.4.1). TSP is the problem of determining whether a path of a particular length (or less), connecting a set of cities, exists or not. 3SAT is the problem of determining whether a set of variable configurations (true/false) exist that render a set of clauses true. And KP is the problem of determining whether there exists a subset of items with differing values and weights exceeding a minimum total value while not exceeding a maximum total weight. All three problems are *decision problems*, that is, problems whose answer is either 'yes' or 'no'. If there exists a configuration of variables such that the solution of the instance is 'yes', the instance is called *satisfiable* and *unsatisfiable* otherwise.

Instances varied in their computational hardness (see Materials and Methods). Both TSP and 3SAT were self-paced (with time limits per trial), while the KP was not. Results for KP have previously been reported elsewhere and are included here for comparison only (Franco et al. 2020).

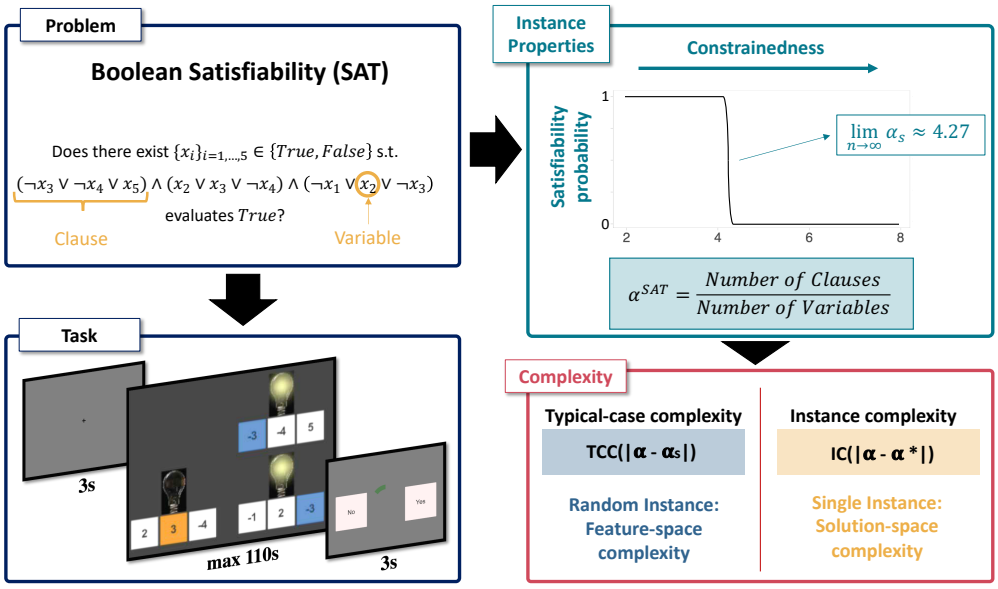
4.2.1 Summary statistics

We first present summary statistics for each of the three tasks. We measured performance as a binary outcome, depending on whether a participant's response was correct or not. Additionally, we studied effort by analyzing time-on-task. This captures another dimension of the agent's problem-solving process that is not entirely determined by performance because, unlike algorithms implemented by electronic computers, humans have the option to stop working independently of the solving strategy. It is worth noting that the effort analysis was not performed for the KP, since this task was not self-paced.

In the TSP, all instances had 20 cities and a time limit of 40 s. The number of cities and time limit were selected, based on pilot data, to ensure that the task was neither too difficult nor too easy (see Materials and Methods). Mean *human performance*, measured as the proportion of trials in which a correct response was made, was 0.85 (min = 0.76, max = 0.93, SD = 0.05). Participants' average time spent on an instance was 32.2 s and ranged from 19.9 s to 39.2 s (SD = 5.2). Performance did not vary during the course of the task, but time-on-task decreased as the task progressed (Appendix C).

All instances of the 3SAT task had 5 variables and a time limit of 110 s. Similar to TSP, the number of variables and time limit were selected, based on pilot data, to target a specific average performance ($\approx$ 85%; see Materials and Methods). Mean *human performance* was 0.87 (min = 0.75, max = 0.98, SD = 0.06). The average

Figure 4.2.1: **3SAT problem, complexity metrics and experimental design. The problem.** The aim is to determine whether a Boolean formula is *satisfiable*. **The task.** The Boolean formula is represented with a set of light bulbs (clauses), each of which has three switches underneath (literals) that are characterized by a positive or negative number. The number on each switch represents the variable number, which can be turned on or off (TRUE or FALSE). The aim is to determine whether there exists a way of turning on and off variables such that all the light bulbs are turned on (formula evaluates TRUE). **Instance properties.** The constrainedness of the problem (α) is captured by the ratio of clauses to variables. This parameter characterizes the probability that a random instance of the problem is satisfiable. In the limit this probability undergoes a phase transition around the satisfiability threshold (α_s). **Complexity metrics.** Instances near this threshold are on average harder to solve than instances further away. Average hardness is captured by the typical-case complexity metric (TCC). This metric can be estimated entirely from the features of the problem (feature-space) without the need to solve the problem. Alternatively, instance complexity (IC) can be estimated from features of the solution-space. IC is characterized as the difference between the constrainedness of the instance (α) and α^*, the maximum number of clauses that can be satisfied normalized by the number of clauses.



time spent on an instance varied from a minimum of 15.9 s to a maximum of 104.3 s (mean = 60.2, SD = 18.7). Similar to TSP, performance did not vary during the course of the task, but participants tended to spend less time on a trial as the task progressed (Appendix C).

In the KP decision task implemented by Franco et al. 2020, all instances had 6 items. This task was not self-paced, that is, participants had exactly 25 seconds to solve each instance and could not skip to the next screen before the time ended. Mean *human performance* was 83.1% (min = 0.56, max = 0.9, SD = 0.08). Like in the other two tasks, performance did not vary during the course of the task (Appendix C).

4.2.2 Feature-space complexity metrics

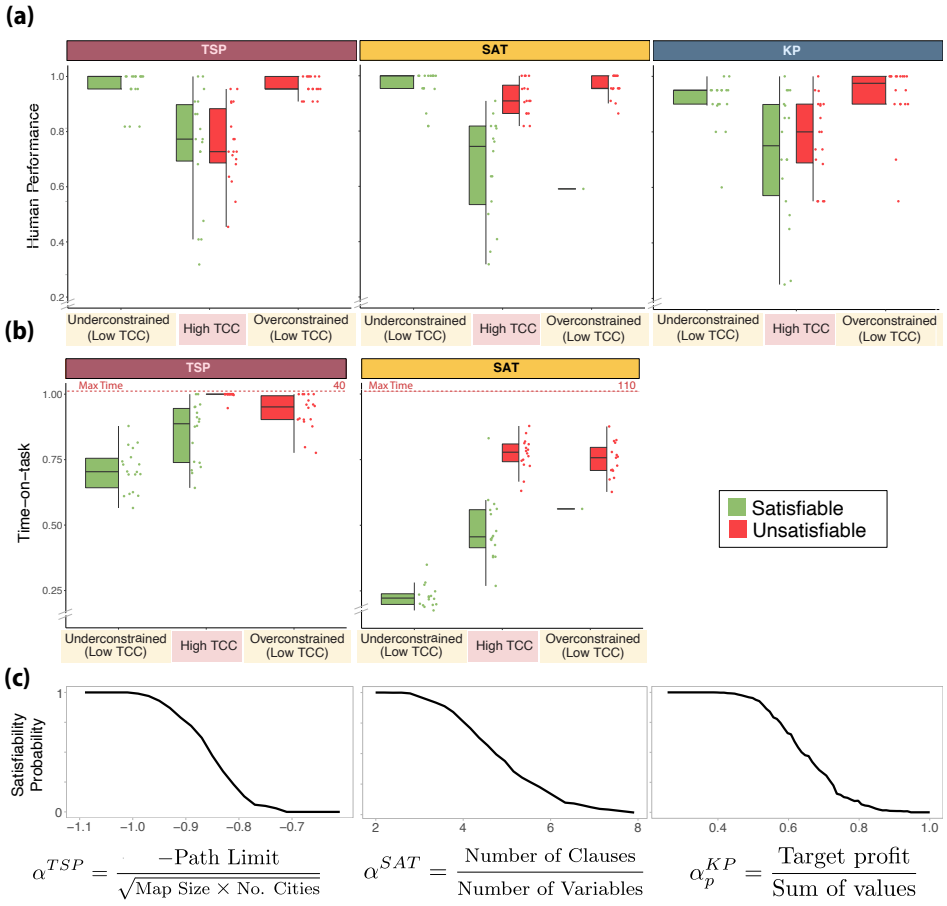
We now examine how generic properties of instances affect the quality of decisions and the computational effort exerted. We study two types of properties: feature-space and solution-space metrics. The main difference between them is that feature-space metrics can be estimated from mathematical properties of the instance without any knowledge of an instance's solution, whereas the calculation of solution-space metrics require knowledge of an instance's solution, that is, require the solution to be computed (Fig 4.2.1).

We first examine the effect of typical-case complexity (TCC), a feature-space metric of complexity, on human performance and effort. This measure is based on a framework in computer science developed to study the drivers of computational hardness in computational problems by analyzing the difficulty of randomly generated instances of those problems. The study of random instances has revealed that there is substantial variation in computational resource requirements for instances with the same input length (Cheeseman, Kanefsky, and Taylor 1991; Percus, Istrate, and Moore 2006; Gent and Walsh 1996; Yadav et al. 2020). This variation in computational hardness has recently been related to various structural properties of instances. In particular, it has been shown for several intractable (specifically, NP-complete) problems, including the KP (Yadav et al. 2020), TSP (Gent and Walsh 1996) and 3SAT (Monasson et al. 1999; Mézard, Parisi, and Zecchina 2002), that there exists a set of parameters $\bar{\alpha}$ that captures the constrainedness of an instance. Moreover, it has been shown that there is a threshold α_s such that random instances with $\alpha \ll \alpha_s$ are mostly satisfiable whereas they are mostly unsatisfiable if $\alpha \gg \alpha_s$. Importantly for our study, it has been shown for each of the problems under consideration that instances near α_s are, on average, computationally harder than instances further away from α_s (Yadav et al. 2020; Gent and Walsh 1996; Cheeseman, Kanefsky, and Taylor 1991; Percus, Istrate, and Moore 2006). In our study, we sampled instances with varying values of α and categorize instances with $\alpha \sim \alpha_s$ as instances with a *high TCC* and instances with $\alpha \gg \alpha_s$ or $\alpha \ll \alpha_s$ as *low TCC* (see Fig 4.2.2 and Materials and Methods).

We first examine the effect of TCC on human performance across problems. We hypothesized that participants would have lower performance on instances with high TCC compared to those with low TCC. We found that this was indeed the case for both TSP and 3SAT as well as for KP (TSP: $\beta_{0.5} = -2.10$, $HDI_{0.95} = [-2.50, -1.73]$, Table F.2 Model 2; 3SAT: $\beta_{0.5} = -1.58$, $HDI_{0.95} = [-1.95, -1.20]$, Table F.1 Model 2; KP: $\beta = -1.327$ $P < 0.001$, main effect of TCC on performance, GLMM; Fig 4.2.2a).

Instances with an $\alpha \gg \alpha_s$ or $\alpha \ll \alpha_s$ are considered to have a low TCC. However, these instances belong to two structurally different regions, namely an overconstrained and an underconstrained region. We studied whether differences in constrainedness affected performance among low TCC instances. We found that for the TSP and 3SAT, there was no difference in performance between underconstrained and overconstrained regions (TSP: $\beta_{0.5} = 0.14$, $HDI_{0.95} = [-0.58, 0.87]$, Table F.2 Model 3; 3SAT: $\beta_{0.5} = -0.43$, $HDI_{0.95} = [-1.12, 0.28]$, Table F.1 Model 3; the difference in effect, $overconstrained - underconstrained$, on performance, GLMM). These results are consistent with those obtained previously in relation to KP ($\beta = 0.250$, $P = 0.355$, the difference in effect, $overconstrained - underconstrained$, on performance, GLMM; Fig 4.2.2a). Taken together, these findings suggest that the mapping

Figure 4.2.2: **Typical-case complexity (TCC). (a) Human performance and satisfiability probability.** Each dot represents an instance of one of the three problems considered. For each instance human performance corresponds to the proportion of participants that solved an instance correctly. The instances are categorized according to their constrainedness region (α) and their TCC. The correct solution (satisfiability) of an instance is represented by its color. **(b) Time-on-task and TCC.** Median time spent solving an instance before submitting an answer. Time is represented as a proportion of the maximum time allotted on each trial (40s in the TSP and 110s in the 3SAT). **(c) Satisfiability probability and constrainedness parameter α.** Probability that a random instance is satisfiable as a function of α (the probability is empirically estimated; see Materials and Methods). In the underconstrained region (low TCC) the satisfiability probability is close to one while in the overconstrained region (low TCC) the probability is close to zero. The region with a high TCC corresponds to a region in which the probability is close to 0.5. *The box-plots represent the median, the interquartile range (IQR) and the whiskers extend to a maximum length of 1.5*IQR*



between α and TCC captures the effect of α on performance.

We also expected TCC to have an effect on time-on-task. We hypothesized that participants would spend more time on instances with high TCC. We found this to be the case for 3SAT and TSP (3SAT: $\beta_{0.5} = 0.149$, $HDI_{0.95} = [0.116, 0.182]$, Table F.3 Model 2; TSP: $\beta_{0.5} = 0.118$, $HDI_{0.95} = [0.090, 0.147]$, Table F.4 Model 2; effect of TCC on time-on-task as a proportion of the maximum possible time, censored linear

mixed-effects models (CLMM), Fig 4.2.2b). The effect was mainly driven by the constrainedness level (α). Specifically, participants spent less time-on-task on instances in the underconstrained region (3SAT: $\beta_{0.5} = -0.352$, $HDI_{0.95} = [-0.385, -0.318]$ Table F.3 Model 3; TSP: $\beta_{0.5} = -0.199$, $HDI_{0.95} = [-0.233, -0.164]$, Table F.4 Model 3; difference in time-on-task between instances in the underconstrained region and those with high TCC ($\alpha \sim \alpha_s$), CLMM). In the TSP, participants spent less time on overconstrained instances compared to those instances with $\alpha \sim \alpha_s$, but this effect was not significant ($\beta_{0.5} = -0.024$, $HDI_{0.95} = [-0.059, 0.011]$, difference in time-on-task between instances in the overconstrained region and $\alpha \sim \alpha_s$, CLMM; Table F.4 Model 3). In contrast, in the 3SAT participants spent more time on overconstrained regions compared to those instances with $\alpha \sim \alpha_s$ ($\beta_{0.5} = 0.071$, $HDI_{0.95} = [0.036, 0.106]$, difference in time-on-task between instances in the overconstrained region and with high TCC, CLMM; Table F.3 Model 3). It is worth noting that in 3SAT, the more constrained the problem is, the higher the amount of clauses presented, which could have driven this effect.

Our results so far show that participants expend more effort on instances with higher TCC and yet they perform worse on these instances. This suggests a negative correlation between time-spent and performance (TSP: $\beta_{0.5} = -0.1$, $HDI_{0.95} = [-0.13, -0.08]$, Table F.2 Model 5; 3SAT: $\beta_{0.5} = -0.02$, $HDI_{0.95} = [-0.02, -0.01]$, Table F.1 Model 5); effect of time-spent on performance, GLMM).

4.2.3 Solution-space complexity metrics

In the previous section, we studied the effects of feature-space complexity metrics on human performance and effort. These metrics can be estimated based on a problem's input, that is, without the need to solve the instance. We now turn our attention to complexity metrics based on an instance's *solution space*. We will use the term solution space to refer to the set of *solution witnesses* of an instance, that is, the set of configurations of variables (e.g., possible paths or variable assignments) that satisfy an instance's constraints. Note that in order to estimate solution-space metrics, the instance, or a harder variant, has to be solved. In some cases, all possible solution witnesses must be found.

An important difference in the structure of instances, is their *satisfiability*, that is, whether the instance's solution is 'yes' or 'no'. We found that satisfiability affects performance but that this effect varies between problems. In 3SAT, participants performed worse on satisfiable instances ($\beta_{0.5} = -1.35$, $HDI_{0.95} = [-1.73, -0.99]$, main effect of satisfiability, GLMM, Table F.1 Model 8), whereas there was no significant effect of satisfiability on performance in the TSP and the KP (TSP: $\beta_{0.5} = -0.06$, $HDI_{0.95} = [-0.34, 0.22]$, Table F.2 Model 6; KP: $\beta_{0.5} = -0.29$, $HDI_{0.95} = [-0.57, 0.01]$, Table F.6 Model 1; main effect of satisfiability, GLMM).

Turning our attention to the effect of satisfiability on time-on-task, we find that less time was spent on satisfiable instances in both TSP and 3SAT (TSP: $\beta_{0.5} = -0.17$, $HDI_{0.95} = [-0.20, -0.15]$, Table F.4 Model 4; 3SAT: $\beta_{0.5} = -0.32$, $HDI_{0.95} = [-0.35, -0.29]$, Table F.3 Model 4; effect of satisfiability on time-on-task, CLMM). We further explored the effect of satisfiability by studying its interaction effect with TCC. We only found an interaction effect between satisfiability and TCC in 3SAT, in relation to both performance and time-on-task (Fig 4.2.2; Appendix A).

In summary, we observed that participants spent less time-on-task on satisfiable

instances in both TSP and 3SAT, yet the effect of satisfiability on performance varied across problems. Moreover, our results suggest that satisfiability and TCC might interact and affect performance and time-on-task on some problems.

We can analyze the drivers of hardness in satisfiable instances at a more granular level by studying the number of solution witnesses of an instance. This generic feature of decision instances captures the constrainedness of an instance: a higher value of witnesses is related to a lower degree of constrainedness. It is worth noting that this metric is only informative for satisfiable instances (by definition, unsatisfiable instances have zero solution witnesses). Thus, we restrict our analysis to these instances.

We found, in line with our hypothesis, a positive effect of the number of witnesses on performance in all three problems (3SAT: $\beta_{0.5} = 0.62$, $HDI_{0.95} = [0.49, 0.79]$; TSP: $\beta_{0.5} = 0.45$, $HDI_{0.95} = [0.37, 0.53]$; KP: $\beta_{0.5} = 0.26$, $HDI_{0.95} = [0.19, 0.34]$; main effect of the number of witnesses on performance, GLMM; Table F.5 Models 1,4,6; Fig 4.2.3a). We further hypothesized that participants would spend less time solving instances with a higher number of witnesses. This was indeed the case (TSP: $\beta_{0.5} = -0.02$, $HDI_{0.95} = [-0.03, -0.02]$, Table F.4 Model 5; 3SAT: $\beta_{0.5} = -0.041$, $HDI_{0.95} = [-0.047, -0.036]$, Table F.3 Model 5; effect of number of witnesses on time-on-task, CLMM; Fig 4.2.3b)). These results suggests that, among satisfiable instances, the more constrained an instance, the harder it is to solve.

It is worth noting that the number of witnesses is a metric conceptually similar to TCC. After all, TCC is a mapping from expected constrainedness (α) to hardness. We studied the link between these metrics and found that the effect of TCC on performance on satisfiable instances is driven, at least partially, by the number of witnesses of an instance (see Appendix D).

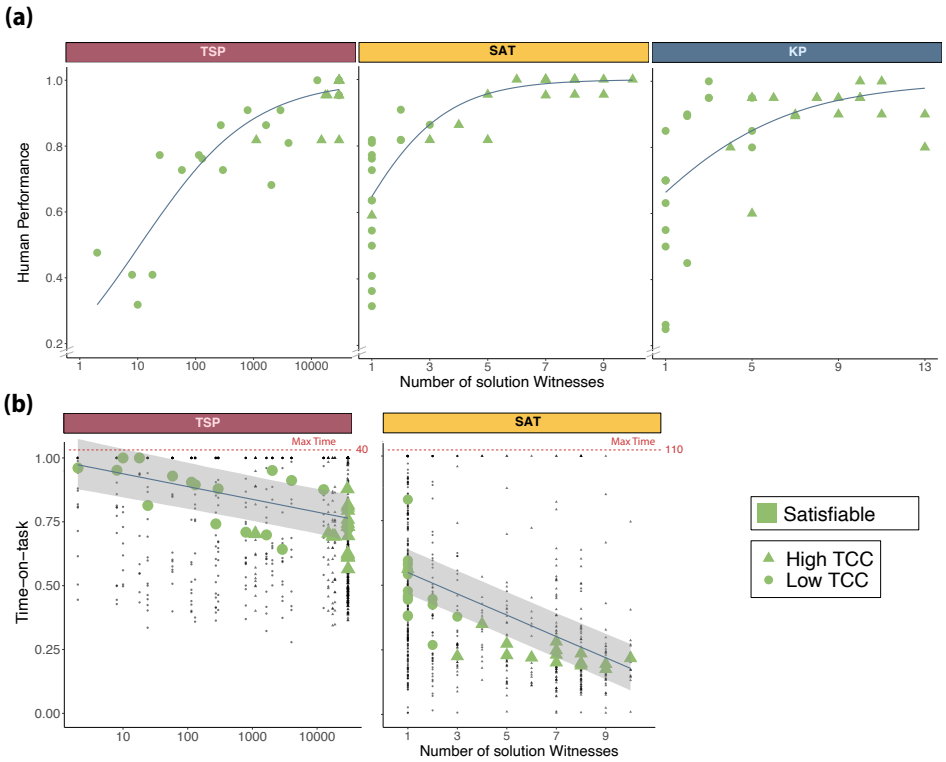
An alternative solution-space complexity metric that can be used to study the difficulty of all instances (both satisfiable and unsatisfiable) is instance complexity (IC) (Franco et al. 2020). It is related to the constrainedness of an instance and the order parameter $\bar{\alpha}$. It is defined based on the distance between the decision threshold of an instance and the maximum attainable value in the optimization variant of the instance. For example, the optimization variant of an instance of the TSP corresponds to finding the minimum path-length connecting all cities. For the KP, it corresponds to finding the maximum value that can fit into the knapsack given the weight constraint. Analogously, for the 3SAT, the optimization version (MAX-SAT) corresponds to finding the maximum number of clauses that can be rendered true simultaneously.

We define IC as the absolute value of the normalized difference between target value of the decision variant and the maximum value attainable of the corresponding optimization variant. In the KP, for example, it is the absolute value of the difference between target profit of the decision instance and the maximum profit attainable of the corresponding optimization instance, divided by the sum of the values of all items, that is,

$$IC_{KP} = \left|\alpha_p - \alpha_p^\star\right| = \left|\frac{\text{Target profit} - \text{Maximum profit attainable}}{\sum v_i}\right|,$$

where the decision instance and the corresponding optimization instance have the same set of items and the same total weight (capacity) constraint. Intuitively, IC in KP is the normalized value of the distance between the target profit of a

Figure 4.2.3: **Number of solution witnesses.** The number of witnesses is defined as the number of *state-space combinations* (i.e., paths, items or switch-setups) that satisfy the constraints. On satisfiable instances, the problem becomes harder as the number of witnesses approaches 0. Only satisfiable instances are included. **(a) Human performance.** Each green shape represents the mean accuracy per instance. The blue line represents the marginal effect of the number of solution witnesses on human performance (GLMM Table F.5 Models 1,4,6). **(b) Time-on-task.** Each green shape represents the median time-on-task per instance. The blue line represents the marginal effect (and 95% credible interval) of the number of solution witnesses on time-on-task (LMM Table F.4 Model 5 and Table F.3 Model 5). Each black dot corresponds to the time-on-task of one participant while solving a single instance.

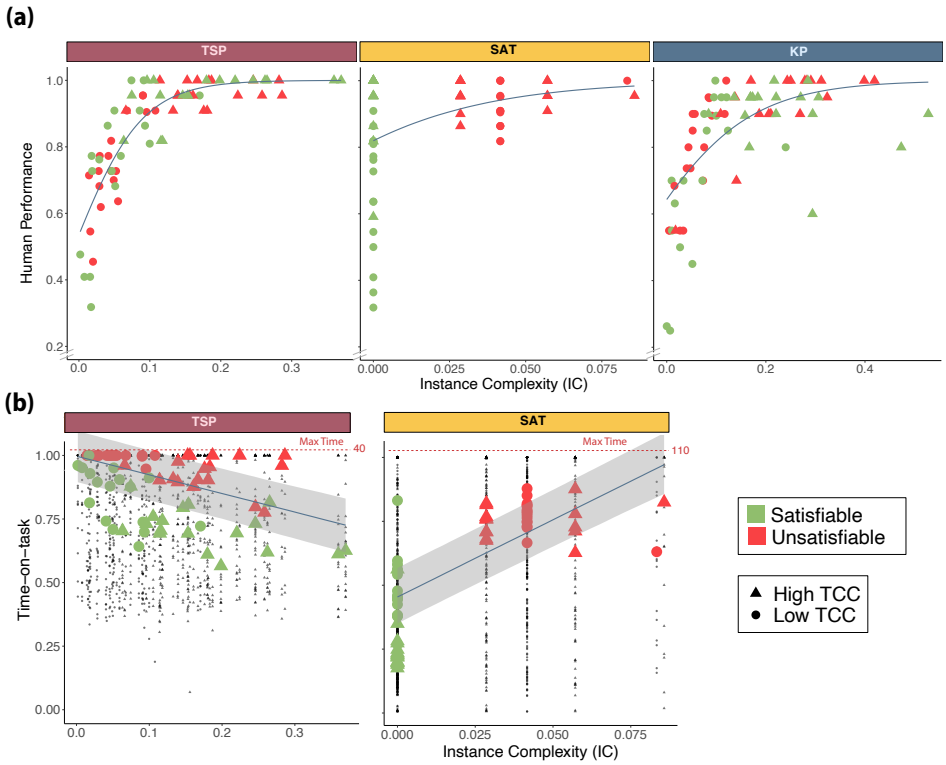


decision instance and the maximum profit that can be attained with the same set of items and the same capacity constraint. The corresponding expressions for TSP and 3SAT are provided in the Methods section.

We studied the effect of IC on performance and effort in each of the problems. Note that lower values of IC indicate that the decision threshold is closer to the optimum, which corresponds to a higher level of computational hardness. Therefore, we expected a positive relation between IC and performance. We found a positive non-linear relation in all problems (KP: $\beta_{0.5} = 9.05$, $HDI_{0.95} = [7.20, 11.02]$, Table F.6 Model 2; TSP: $\beta_{0.5} = 21.13$, $HDI_{0.95} = [17.63, 24.91]$, Table F.2 Model 7; 3SAT: $\beta_{0.5} = 30.30$, $HDI_{0.95} = [21.95, 39.24]$, Table F.1 Model 6; the effect of IC on performance, GLMM; Fig 4.2.4a).

IC is a metric at the level of individual instances and thus we expected that it captures a substantial amount of the variability in performance between instances. Indeed, IC was able to explain a high proportion of the variance in average instance

Figure 4.2.4: **Instance complexity.** Instances become harder as $IC = |\alpha_p - \alpha_p^\star|$ approaches 0. **(a) Human performance.** Green and orange shapes represent the mean accuracy for each instance. The blue lines represents the marginal effect of IC on human performance (GLMM Table F.6 Model 2, Table F.2 Model 7, Table F.1 Model 6). **(b) Time-on-task.** Green and orange shapes represent the median time-on-task for each instance of the TSP and 3SAT problems. The blue lines represents the marginal effect (and 95% credible interval) of IC on time-on-task (LMM Table F.4 Model 6, Table F.3 Model 6). Each black dot corresponds to the time spent by a single participant on a particular instance.



performance in the TSP and the KP (KP: $R^2 = 0.65$; TSP: $R^2 = 0.75$) but was lower in 3SAT (3SAT: $R^2 = 0.16$). We explore this further in the Appendix E.

Next, we explored how well IC predicted time-on-task. We expected a negative relation between IC and the average time spent on an instance. This was the case for TSP ($\beta_{0.5} = -0.735$, $HDI_{0.95} = [-0.901, -0.581]$, main effect of IC on time-on-task, CLMM; Table F.4 Model 6; Fig 4.2.4b), but for the 3SAT we found a significant positive effect ($\beta_{0.5} = 6.04$, $HDI_{0.95} = [5.41, 6.70]$, main effect of IC on time-on-task, CLMM; Table F.3 Model 6; Fig 4.2.4b). Based on this result, we hypothesized that the positive effect of IC on time-on-task in 3SAT could have been driven by the effect of satisfiability, but we are unable to test this hypothesis directly. Therefore, we investigated the effect of IC on time-on-task in unsatisfiable instances only and found a non-significant negative effect ($\beta_{0.5} = -0.557$, $HDI_{0.95} = [-1.912, 0.782]$, main effect of IC on time-on-task for unsatisfiable instances, CLMM; Table F.3 Model 7). These results indicate a negative relation between IC and time-on-task in the TSP, whereas in 3SAT the results are inconclusive.

We have shown that generic instance-level complexity metrics are able to explain differences in performance and time-on-task across instances and problems.

However, it remains an open question whether these generic properties can shed light on how humans solve those problems. To explore this question, we investigated whether our metrics could explain differences in the number of clicks across instances. The number of clicks is a useful metric in studying the algorithms implemented by humans. Specifically, the number of clicks is related to the way that the problem's state space is explored. In the 3SAT, the state space consists of all possible on-off switch setups (2^5 possible combinations) while in the TSP the state space consists of all possible ordered path selections ($2^{\binom{20}{2}} = 2^{190}$ possible combinations). Arguably, participants search the state space by clicking on different state combinations in order to decide whether an instance is satisfiable or not. Differences in the quantity of clicks used to solve an instance can shed light into how the state space is explored (under the assumption that the state space is explored by clicking on elements in the task). We investigated whether generic properties of the instance captured differences in the number of clicks.

We found that the length of search in the state space, that is, the set of paths or variable configurations, is related to two properties of the instance, namely satisfiability and complexity. Search was longer in general in the case of unsatisfiable instances and there was a positive effect of TCC on search length. Moreover, longer search was also related to lower values of IC and lower number of witnesses (see Appendix B). These patterns suggest that the length of search in the state space can be explained, at least partially, by the properties of an instance. Interestingly, the effect of our metrics on performance can shed light on the possible strategies used by participants (see Discussion).

4.3 Discussion

Human behavior arises as an interaction between the agent, subject to limited cognitive capacities, and its environment (Herbert A Simon 1990). Much research on this interaction to date has focused in characterizing cognitive strategies employed by agents in a given environment (Tversky and Kahneman 1974; Todd and Gerd Gigerenzer 2012; Gerd. Gigerenzer and Selten 2001). Comparatively little work has investigated how properties of the task environment relate to cognitive demands and how these interact with cognitive capacities. In the present study, we propose a generic framework for quantifying computational hardness of cognitive tasks based on structural properties of individual instances of the the underlying computational problem. We find that a set of metrics based on these properties predict both task performance and effort exerted across three cognitive tasks related to different NP-complete computational problems.

More specifically, using a controlled experiment, we show that three generic properties of NP-complete problems, typical-case complexity (TCC), the number of solution witnesses, and instance complexity (IC), affect human performance and effort exerted when performing a task. While the extent of effort increased with higher complexity of instances, efficacy, and thus performance in those instances, decreased. We show that the relation between the complexity metrics presented on the one hand and task performance and effort exerted on the other, are similar across three different NP-complete problems.

Our results complement findings from computer science and suggest that hard-

ness stems partially from intrinsic difficulty of the problem and the instance, regardless of the algorithm and the computing device used. In particular, our findings suggest that the same intrinsic hardness metrics describe the performance of algorithms executed by both electronic computers (Gent and Walsh 1996; Cheeseman, Kanefsky, and Taylor 1991; Percus, Istrate, and Moore 2006; Monasson et al. 1999; Mézard, Parisi, and Zecchina 2002; Yadav et al. 2020) and humans. This is particularly interesting because the theory in which our analysis is based is derived without taking into account limits on human computation. For instance, no memory constraints are imposed on the solving algorithms. Interestingly, our results also show that computational hardness affects how much time an agent decides to spend on an instance. This is far from obvious because, unlike the standard algorithms executed by electronic computers, humans have the option to stop working independently of the solving strategy.

Critically, our results provide support for the premise that a comprehensive and accurate characterization of human behavior requires the study of both 'blades' of the scissors: an agent's cognitive capacities as well as the task environment. The proposed approach can shed light on how to operationalize bounded rationality (Herbert A Simon 1990) by shaping the canvas to which cognition must be confined in order to model a computationally feasible agent.

4.3.1 Computational complexity in cognition

The role of computational complexity in cognition has been studied before. Problem complexity has been used to study the limits of what is potentially human computable (van Rooij et al. 2019; Frixione 2001; Tsotsos 1990). According to this work, many tasks we face in our lives—and corresponding computational models of human behavior—are computationally intractable (NP-hard) (van Rooij et al. 2019), including planning, learning and many forms of reasoning (for example, analogy, abduction and Bayesian inference) (van Rooij et al. 2019). This means that the computational requirements quickly grow to levels that make solving those tasks infeasible within a reasonable amount of time and memory.

This analysis is, however, too coarse to explain differences in performance and behavior across the class of human-computable problems. Such differences have generally been ascribed to the solver or the agent (Murawski and Bossaerts 2016; Bourgin et al. 2017; Shepard and Metzler 1971; Dry et al. 2006; Guid and Bratko 2013; Stazyk, Ashcraft, and Hamann 1982; De Visscher and Noël 2014). This approach, however, is problematic given the diversity of algorithms implemented and their specificity to a particular problem (Ohlsson 2012).

We propose that a new level of analysis be included in the study of cognition: instance-level complexity. This additional level of analysis describes the generic or intrinsic complexity of problems at a more granular level. In the present study, we show that our conceptual approach captures differences in behavior across different NP-complete problems without reference to an algorithm or particular computational device. More specifically, we explored the effect of three generic complexity metrics on human performance. Each of them can be used to unearth generalities in human behavior. Typical-case complexity (TCC) captures the average hardness of a random ensemble of instances of a problem based on its constrainedness. Critically, TCC can be computed ex-ante—without knowledge of an instance's solution.

Instance complexity (IC) maps constrainedness to complexity, but does this at the level of a single instance rather than an average across instances. Finally, the number of solution witnesses captures a structural property of an instance that is related to the hardness of search for satisfiable instances.

Our three metrics capture generalities in behavior using generic metrics of computational hardness on NP-complete problems, just like metrics of uncertainty, such as mean, variance and other statistics, capture generalities in behavior in probabilistic problems. Importantly, our framework can be applied to other decision problems in classes P or NP (Cheeseman, Kanefsky, and Taylor 1991; Percus, Istrate, and Moore 2006; Monasson et al. 1999), and has also been shown that it can be extended to optimization problems (Franco et al. 2020).

The generality of TCC is limited by its dependence on a particular sampling distribution. We sampled instances for each of the problems from a specific procedure in which the components of the instances were randomly sampled from uniform distributions. We leave it to future research to study whether TCC can be extended to other probability distributions, and particularly, to those found in real life (Bogdanov and Trevisan 2006).

Importantly, we provided two alternatives to TCC (IC and number of witnesses), which do not depend on a sampling procedure. These metrics quantify the hardness of specific instances of problems. However, they do come at a cost: these metrics are computationally intensive. That is, in order to compute them, the decision problem, or a harder variant, needs to be solved first. For IC to be estimated, the optimization variant of the instance needs to be solved, whereas to compute the number of witnesses, all of the possible witnesses of an instance need to be counted.

We argue that the computational requirements of calculating these metrics is not prohibitive in the context of the study of human problem-solving and cognition in general. These metrics can be used to predict generalities in human behavior with the aid of any of the resources at hand, including electronic computers. Therefore, since the practical instances of problems solvable by humans are relatively small compared to those solvable by electronic computers, cognitive scientists effectively have access to an oracle machine to estimate computationally intensive metrics.

4.3.2 Future directions

This paper focuses on generalities across problems within a well-defined class (i.e., NP-complete). A related question is whether intrinsic characteristics specific to a problem can complement the generic metrics presented here. Intrinsic metrics of complexity, specific to a problem, have been previously shown to affect performance. Specifically, for all three problems considered in this study, measures derived from the features of the problem have been shown to affect computational time of algorithms executed on electronic computers (Smith-Miles and Lopes 2012; Hill and Reilly 2000; Van Hemert 2005; Nudelman et al. 2004). Additionally, problem-specific complexity metrics have been show to be related to human performance in the optimization variants of the TSP (MacGregor and Chu 2011; Hirtle and Gärling 1992). Future work should be undertaken to study how instance-complexity generic metrics and problem-specific measures jointly affect human performance.

Importantly, our results suggest that the metrics put forward in this study are generic as they provide both ex-ante and ex-post predictability across different prob-

lems. However, our work also highlights that certain structural properties of the problem might have problem-specific effects that could interact with the effect of generic metrics of hardness. This is particularly evident in the 3SAT. In this task we find that IC explains less of the variance in performance than in the other two tasks and that the effect of IC on time-on-task is inconclusive. This might be related to the intertwinement of satisfiability and IC in this problem. Specifically, the structure of the 3SAT problem generates an unavoidable confounding between these two metrics given that all satisfiable instances have $IC = 0$, thus rendering IC incapable of explaining variance across satisfiable instances. This is further relevant because in this task, unlike in the other two, satisfiability has a significant effect on performance. Taken together, this suggests that the effect of IC on performance in the 3SAT might be incongruously driven by satisfiability, in a way that cannot be differentiated in our experimental design. In future studies, it should be attempted to disentangle these effects, for example, by studying the related maximum satisfiability problem (MAX-SAT). More importantly, these results warrant further investigation of the effect of satisfiability, and other structural properties, on human behavior. Moreover, future work could explore the differences in these effects across classes of problems. For instance, NP-complete problems could be categorized into finer classes based on the effect of particular properties on human problem-solving. Our findings would suggest that more abstract logical problems might be solved differently to other more life-pertinent problems, such as KP and TSP.

We investigated the effect of different metrics of instance-level complexity keeping the size of instances fixed. An additional dimension in this framework that has been shown to affect human behavior is an instance's size (Dry et al. 2006; MacGregor and Chu 2011) and the size of the state space (that is, the number of possible combinations or paths) (van Opheusden and Ma 2019; Murawski and Bossaerts 2016). Additionally, the instance complexity metrics we presented are based on the satisfiability threshold and the number of witnesses. Recently, it has been shown that the performance of algorithms, designed for electronic computers, as α approaches α_s, is not only related to the decrease in the number of witnesses, but also to the shattering of witnesses into distinct clusters (Budzynski, Ricci-Tersenghi, and Semerjian 2019; Krzakala et al. 2006). Further research is needed to integrate these different dimensions of complexity and determine their combined effect on human problem-solving.

We have argued that the framework presented here can be used to characterize the effect of the task environment on human performance. However, instance-level complexity metrics can also shed light on the type of strategies employed by agents. Note, for example, that our results suggest that participants did not predominantly perform random search. In a random algorithm, random combinations from the *state space* (i.e., paths or variable configurations) are tried and an answer (yes/no) is selected depending on whether a solution witness is found. If participants implemented such an algorithm, we would expect the length of search to be similar on all unsatisfiable instances given that completing an exhaustive search of the state space is unlikely because of the limits on time and number of clicks. This, however, is not what we found. In unsatisfiable instances the length of search, time-on-task (in TSP) and performance were affected by IC. In fact, by applying the same argument, we can rule out other more directed search heuristics such as greedy algorithms (Ausiello et al. 1999), which have been proposed to be linked to human

behavior (Murawski and Bossaerts 2016). Overall, our results suggest that when people solve decision problems, they implement procedures to exclude alternatives from the witness solution set. If this was not the case, we would not find any effect of our complexity metrics among unsatisfiable instances. Further research is needed in order to disentangle between prospective algorithms and explore how instance-level complexity measures can be used to inform the study of algorithm selection in humans.

* * *

We provide empirical evidence that studying the intrinsic computational hardness of the task environment predicts human cognitive effort and performance on a task. This has important practical implications, which could help improve human decision-making. The approach presented here could be used to quantify the computational hardness of problems people face in everyday life, such as making financial investments or health insurance decisions. Our generic approach would provide a rigorous method to estimate average quality of such decisions. Both designers of products as well as regulators could use a framework like ours to identify upper limits in the complexity of the tasks that consumers face when dealing with those products and services.

4.4 Materials and Methods

4.4.1 Ethics statement

The experimental protocol was approved by the University of Melbourne Human Research Ethics Committee (Ethics ID 1749594.2). Written informed consent was obtained from all participants prior to commencement of the experimental sessions. Experiments were performed in accordance with all relevant guidelines and regulations, including the Declaration of Helsinki.

4.4.2 Participants

A total of 47 participants were recruited in two separate groups from the general population (group 1: 24 participants; 12 female, 12 male; age range = 19-35 years, mean age = 24.1 years; group 2: 23 participants; 13 female, 10 male; age range = 18-32 years, mean age = 23.3 years). Inclusion criteria were based on age (minimum=18 years, maximum=35 years) and normal or corrected-to-normal vision.

Each group of participants were asked to solve a set of random instances of a computational problem. Group 1 participants were presented with 64 instances of the Boolean satisfiability problem (3SAT). Group 2 participants were presented with 72 instances of the decision variant of the traveling salesperson problem (TSP). Some trials and participants were excluded due to different issues (see section 4.4.5).

4.4.3 Experimental tasks

Boolean satisfiability task

This task is based on the 3-satisfiability problem. In this problem, the aim is to determine whether a boolean formula is *satisfiable*. In other words, given a propositional

formula, the aim is to determine whether there exists at least one configuration of the variables (which can take values TRUE or FALSE) such that the formula evaluates to TRUE. The propositional formula in 3SAT has a specific structure. Specifically, the formula is composed of a conjunction of clauses that must all evaluate TRUE for the whole formula to evaluate TRUE. Each of these clauses, takes the form of an OR logical operator of three literals (variables and their negations). An example of a 3SAT problem is:

$$\boxed{\begin{array}{c}\text{Does there exist } x_i \in \{TRUE, FALSE\} \text{ s.t.} \\ (\neg x_3 \vee \neg x_4 \vee x_5) \wedge (x_2 \vee x_3 \vee \neg x_4) \wedge (\neg x_1 \vee x_2 \vee \neg x_3) \\ \text{evaluates } TRUE?\end{array}}$$

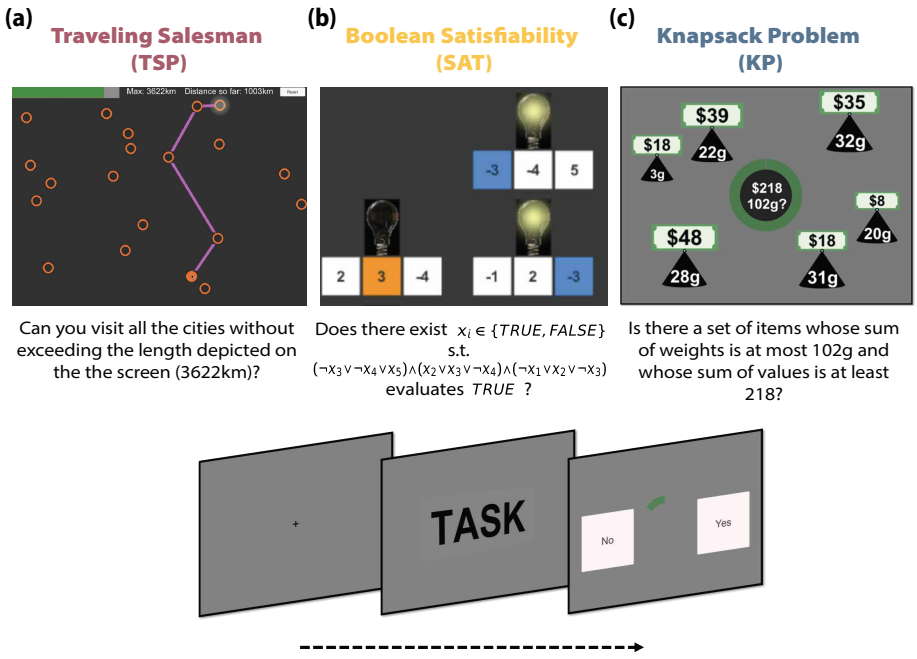
In order to represent this in an accessible way to participants we developed a task composed of switches and light bulbs (Fig 4.4.1b). Participants were presented with a set of light bulbs (clauses), each of which had three switches underneath (literals) that were represented by a positive or negative number. The number on each switch represented the variable number, which could be turned on or off (TRUE or FALSE). The aim of the task is to determine whether there exists a way of turning on and off variables such that all the light bulbs are turned on (that is, the formula evaluates TRUE).

At the beginning of each trial, participants were presented with a different instance of the 3SAT problem. A bar in the top-right corner of the screen indicated the time remaining in the trial. Each participant completed 64 trials (4 blocks of 16 trials with a rest period of 60 seconds between blocks). Trials were self-paced with a time limit of 110 seconds. Participants could use the mouse to click on any of the variables to select their value ($\{blue = TRUE, orange = FALSE\}$). A light bulb above each clause indicated whether a clause evaluated to TRUE (light on) given the selected values of the variables underneath it. The number of clicks in each trial was limited to 20. The purpose of this limit was to discourage participants from using a trial-and-error strategy to solve the instances. When participants were ready to submit their solution, they pressed a button to advance from the screen displaying the instance to the response screen where they responded YES or NO. The time limit to respond was 3 seconds, and the inter-trial interval was 3 seconds as well. The order of instances and the side of the YES/NO button on the response screen were randomized for each participant.

Instance sampling

A random instance is a selection of clauses and literals in which M clauses of three literals are chosen randomly. Each of the literals is associated with one of N variables. Both numerical (Monasson et al. 1999) and analytical (Mézard, Parisi, and Zecchina 2002) evidence suggests that in the limit $N \to \infty$, there exists a value of the clause to variables ratio $\alpha = M/N$, α_s^{SAT}, such that typical instances are satisfiable for $\alpha < \alpha_s^{SAT}$, while typical instances are unsatisfiable for $\alpha > \alpha_s^{SAT}$. The current best estimate for the satisfiability threshold, α_s^{SAT}, as $N \to \infty$ is 4.267 (Mézard, Parisi, and Zecchina 2002) (note that the value of α_s is a function of the number of literals per clause, which was fixed at 3 in this study). As $N \to \infty$, instances near the threshold are on average harder to solve (Cheeseman, Kanefsky, and Taylor 1991; Percus, Istrate, and Moore 2006). We exploit both the threshold phenomenon in satisfiability and its link to computational hardness.

Figure 4.4.1: **Experimental Tasks. (a) Traveling salesperson task.** Participants are given a list of cities displayed on a rectangular map on the screen and a limit L on path length. The problem is to determine whether there *exists* a path connecting all N cities with a distance at most L. The task was interactive. Participants could click from city to city and the corresponding path and distance traveled would display and update automatically. This stage lasted a maximum of 40 seconds. Afterwards, participants had 3 seconds to make their response. **(b) Boolean satisfiability task.** In this task, the aim is to determine whether a Boolean formula is *satisfiable*. The Boolean formula is represented with a set of light bulbs (clauses), each of which has three switches underneath (literals) that are characterized by a positive or negative number. The number on each switch represents the variable number, which can be turned on or off (TRUE or FALSE). The aim of the task is to determine whether there exists a way of turning on and off variables such that all the light bulbs are turned on (the corresponding Boolean formula evaluates to TRUE). The task was interactive. Participants could click on switches to turn them on and the corresponding literals and light bulbs would change color automatically. This stage had a time limit of 110 seconds. Afterwards, participants had 3 seconds to make their response (either a 'YES' or 'NO'). **(c) Knapsack decision task.** Participants are presented with a set of items with different values and weights. Additionally, a capacity constraint and target profit are shown at the center of the screen. The aim is to ascertain whether there exists a subset of items for which (1) the sum of weights is lower or equal to the capacity constraint and (2) the sum of values yields at least the target profit. The task was not interactive. This stage lasted for 25 seconds. Finally, participants had 2 seconds to make their response.



We generated random instances with different degrees of complexity by varying α. We picked a value of α, starting at the lower bound of its range and incrementing in steps of 0.1 until the upper bound was reached. For each value of α, we computed the number of clauses M by multiplying α and a fixed value of N and rounding to

the nearest integer. N (and the time limit for the task) was determined before hand using pilot data to ensure that the task was not too easy nor too hard for participants (i.e. to ensure sufficient variation in performance). Importantly, N was also restricted to values in which the corresponding number of clauses could fit in the screen of the task. Specifically, we restricted the number of clauses to be at most 36.

Once N was fixed, we generated 1000 random instances for each value of M. Each random instance was generated by first selecting the literals for each clause. Each literal is represented by a positive or negative sign (negation of a variable) and is sampled from the set $\{-1, +1\}$ with equal probability. Afterwards, three variables were selected for each clause by sampling without replacement from the set of N variables (Cheeseman, Kanefsky, and Taylor 1991; Monasson et al. 1999).

From the randomly generated instances we first determined the satisfiability threshold of our finite instances ($N = 5$). That is, we calculated the value of α at which half of the randomly generated instances were satisfiable and half were unsatisfiable. This was the case for $\alpha = 4.8$. Based on this we selected a subset of random instances to use in the task.

We asked participants to solve a set of instances randomly sampled from three different regions: an underconstrained region ($\alpha \ll \alpha_s^{SAT}$), a region around the satisfiability threshold ($\alpha \sim \alpha_s^{SAT}$) and an overconstrained region ($\alpha \gg \alpha_s^{SAT}$). Instances near the satisfiability threshold are defined to have a *high TCC*, whereas instances further away from the satisfiability threshold (in the under-constrained or over-constrained regions) are defined to have a *low TCC*. We selected 16 instances from the underconstrained region ($\alpha = 2$) and 16 instances from the overconstrained region ($\alpha = 7$). We then sampled 32 instances near the satisfiability threshold ($\alpha = 4.8$), such that 16 of the selected instances were satisfiable and 16 were not satisfiable.

In order to also ensure a sufficient degree of variability between instances near the satisfiability threshold, we added an additional constraint in the sampling. For each set of instances (satisfiable and not satisfiable) we forced half to have algorithmic complexity less than the median algorithmic complexity at this value of α, and the other half to be harder than the median. The algorithmic complexity was estimated using an algorithm-specific ex-post complexity measure of a widely-used algorithm (*Gecode* propagations parameter). Gecode is a generic solver for constraint satisfaction problems that uses a constraint propagation technique with different search methods, such as branch-and-bound. We chose an output variable, the number of propagations, that indicates the difficulty for the algorithm of finding a solution and whose value is highly correlated with computational time. We did not use compute time directly as a measure of complexity because for instances of small size, like the ones used in this study, compute time is highly confounded with overhead time. Thus, our set of instances in the region $\alpha \sim \alpha_s$ comprised 8 instances in each of the following categories $\{\text{satisfiable}, \text{unsatisfiable}\} \times \{\text{low/high algorithmic difficulty}\}$.

Traveling salesperson task

This task is based on the traveling salesperson problem. Given a set of N cities displayed on a rectangular map on the screen and a limit L on path length, the decision problem is to answer whether there *exists* a path connecting all N cities with a distance of at most L (Fig 4.4.1a).

In the TSP task, each participant completed 72 trials (3 blocks of 24 trials with a rest period of 30 seconds between blocks). Each trial presented a different instance of TSP. Trials were self-paced with a time limit of 40 seconds. Participants could use the mouse to trace routes by clicking on the dots indicating the different cities. The length of the selected route at each point in time was indicated at the top of the screen (together with the maximum route length of the instance). When participants were ready to submit their answer, they pressed a button to advance from the screen displaying the cities to the response screen where they responded YES or NO. The time limit to respond was 3 seconds, and the inter-trial interval was 3 seconds as well. The order of instances and the sides of the YES/NO button on the response screen were randomized for each participant.

Instance sampling

A TSP instance is a collection of N cities, a matrix of distances $\boldsymbol{d}$ between each pair of cities, and a limit L on path length. Here, we restrict the problem to the euclidean TSP; that is, we constraint our distance matrices $\boldsymbol{d}$ to those that can be represented in a two-dimensional map of area M^2.

Just like for 3SAT, it has been proposed that there exists a parameter α^{TSP} that captures the constrainedness of the problem, specifically $\alpha^{TSP} = -L/(M\sqrt{N})$ (Gent and Walsh 1996). Evidence suggests that in the limit $N \to \infty$, there exists a value of α, α_s^{TSP}, such that typical instances are satisfiable for $\alpha \ll \alpha_s^{TSP}$, while typical instances are unsatisfiable for $\alpha \gg \alpha_s^{TSP}$. α_s^{TSP} for the euclidean TSP is estimated at -0.7124 ± 0.0002 in the limit $N \to \infty$ (D. S. Johnson, McGeoch, and Rothberg 1996; Gent and Walsh 1996). As $N \to \infty$ instances near α_s^{TSP} have been shown to be, on average, harder to solve (Gent and Walsh 1996). We use the this insight to vary typical-case complexity of finite instances.

Instances of the TSP had $N = 20$ cities. This value, and the time limit for the task, were determined using pilot data to ensure that the task was not too easy nor too hard for participants (i.e. to ensure sufficient variation in performance). Random instances of the euclidean TSP were then generated by choosing (x,y) coordinates for each of the $N = 20$ cities, uniformly at random from a square with side length $M = 1000$ (Gent and Walsh 1996). We generated 100 sets of coordinates; that is, 100 distance matrices $\boldsymbol{d}$. For each distance matrix, we generated instances with different values of L. We did this by varying the value of α, which was incremented in the range $[-0.25, -1.25]$ with step size 0.02.

To determine the location of the satisfiability threshold in our sample of random instances (with $N = 20$), we determined the value of α at which half of the randomly generated instances were satisfiable and half were unsatisfiable. The satisfiability threshold was located at $\alpha^{TSP} = -0.85$. We randomly sampled instances at this value of α such that half of the selected instances were satisfiable and half were not satisfiable. We also ensured that half of the instances had a number of propagations above the median and half of them had a number of propagations below the median (see description of 3SAT above for details). Thus, our set of of instances in the region $\alpha \sim \alpha_s$ comprised 9 instances in each of the four following categories: $\{\text{satisfiable}, \text{unsatisfiable}\} \times \{\text{low/high algorithmic difficulty}\}$.

For the underconstrained region, $\alpha \ll \alpha_s$, we randomly chose 18 instances from the set of 100 randomly generated instances with $\alpha^{TSP} = -0.99$. For the over-constrained region, $\alpha \gg \alpha_s$, we randomly chose 18 instances from the set of 100 randomly generated instances with $\alpha^{TSP} = -0.71$. We made sure that no two

instances in our set of selected instances had the same set of city coordinates.

Knapsack task

In this paper we report on the experimental data collected on the knapsack decision task by Franco et al. 2020. Their statistical results were used when available.

The knapsack task is based on the 0-1 knapsack problem (KP). An instance of this problem consists of a set of items $I = \{1, \ldots, N\}$ with weights $\langle w_1, \ldots, w_N \rangle$ and values $\langle v_1, \ldots, v_N \rangle$, and two positive numbers c and p denoting the capacity and profit constraint (of the knapsack). The problem is to decide whether there exists a set $S \subseteq I$ such that $\sum_{i \in S} w_i \leq c$, that is, the weight of the knapsack is less than or equal to the capacity constraint; and $\sum_{i \in S} v_i \geq p$, that is, the value of the knapsack is greater than or equal to the profit constraint.

In their study they implemented the knapsack decision problem in the form of the task presented in Fig 4.4.1c. In their task all instances had 6 items ($N = 6$) and w_i, v_i, c and p were integers. In the task each participant completed 72 trials (3 blocks of 24 trials with a rest period of 60s between blocks). Each trial presented a different instance of the KP. Trials had a time limit of 25 seconds and were *not* self-paced. A green circle at the center of the screen indicated the time remaining in each stage of the trial. During the first 3 seconds participants were presented with a set of items of different values and weights. Then, both capacity constraint and target profit were shown at the center of the screen for the remainder of the trial (22 seconds). No interactivity was incorporated into the task; that is, participants could not click on items. When the time limit was reached, participants were presented with the response screen where they responded YES or NO. The time limit to respond was 2 seconds, and the inter-trial interval was 5 seconds. The order of instances and the sides of the YES/NO button on the response screen were randomized for each participant.

Instance sampling

It has been proposed that there exists a set of parameters $\bar{\alpha}^{KP} = (\alpha_c^{KP}, \alpha_p^{KP})$ that captures the constrainedness of the problem, specifically $\alpha_p^{KP} = p / \sum_{i=1}^{N} v_i$ and $\alpha_c^{KP} = c / \sum_{i=1}^{N} w_i$ (Yadav et al. 2020). These parameters characterize where typical instances are generally satisfiable (under-constrained region), where they are unsatisfiable (over-constrained region) and where the probability of satisfiability is close to 50% (satisfiability threshold). Instance near the satisfiability threshold have been shown to be, on average, harder to solve (Yadav et al. 2020).

Instances in Franco et al. 2020 were selected such that α_c^{KP} was fixed ($\alpha_c^{KP} \in [0.40, 0.45]$) and the instance constrainedness varied according to α_p^{KP}. 18 instances were selected from the under-constrained region ($\alpha_p \in [0.35, 0.4]$; *low TCC*) and 18 from the over-constrained region ($\alpha_p \in [0.85, 0.9]$; *low TCC*). Additionally, 18 satisfiable instances and 18 *unsatisfiable* instances were sampled near the satisfiability threshold ($\alpha_p \in [0.6, 0.65]$; *high TCC*).

Like for 3SAT and TSP, high TCC instances were selected such that they varied according to the number of propagations (see description of 3SAT sampling for details).

Procedure

After reading the plain language statement and providing informed consent, participants were instructed in the task and completed a practice session. Each experimental session lasted around 110 minutes. The tasks were programmed in Unity3D (*Unity 3D* 2017) and administered on a laptop.

Participants received a show-up fee of AUD 10 and additional monetary compensation based on performance. In the 3SAT, they additionally received AUD 0.6 for each correct instance submitted plus a bonus of AUD 0.31 per instance if all instances in the task were solved correctly. In the TSP, participants received 0.3 per correct instance submitted plus 0.14 per instance if all instances were solved correctly. In the KP task (Franco et al. 2020), participants received a show-up fee of A$10 and earned A$0.7 for each correct answer.

Note that the 3SAT and TSP tasks were self-paced (with time limits per trial), whereas the KP was not.

4.4.4 Derivation of metrics

We estimated a collection of metrics based on the features of each instance and its solution space. We estimated one feature-space metric and several solution-space metrics. We first defined Typical-case complexity (TCC) according to the problem-parameter α for each task. Estimation of this metric is tightly related to the instance sampling procedure and its derivation is described in the previous section. Instances were sampled such that there was an equal number of instances with low and high TCC on each of the problems.

Once instances for the tasks were sampled, we estimated their solution-space metrics. We estimated the number of solution witnesses for 3SAT instances using exhaustive search and used the Gecode algorithm (Gecode Team 2006) for TSP instances. For the TSP, we allowed the algorithm to stop after finding 30,000 solution witnesses. This was done to reduce the computational requirements of solving an instance. 15 TSP instances reached the 30,000 maximum imposed. Given the variability in the number of witnesses in the TSP, the results on number witnesses are reported in logarithmic scale (natural logarithm).

We define the instance complexity metric (IC) as the absolute value of the normalized difference between target value of the decision variant and the maximum value attainable of the corresponding optimization variant. In the KP, the optimization variant's problem is to find the maximum value attainable given the weights, values and capacity. In the TSP, the optimization variant is to minimize the path traveled given a distance matrix. In the 3SAT, the optimization variant (MAXSAT) is to find the maximum number of satisfiable clauses given the Boolean formula presented. Explicitly, IC is defined as follows:

$$IC_{KP} = \left|\alpha_p - \alpha_p^\star\right| = \left|\frac{\text{Target profit} - \text{Maximum profit attainable}}{\sum v_i}\right|$$

$$IC_{TSP} = \left|\alpha - \alpha^\star\right| = \left|\frac{\text{Path limit - Minimum path}}{\sqrt{\text{Map area} \times \text{Number of cities}}}\right|$$

$$IC_{SAT} = \left|\alpha - \alpha^\star\right| = \left|\frac{\text{Number of clauses - Max number of clauses set to TRUE}}{\text{Number of variables}}\right|$$

In order to estimate the instance complexity metric (IC), the optimization variant of each instance needs to be solved. These optima were estimated using Gecode (Gecode Team 2006) in TSP and using the *RC2* algorithm from the 'pysat' python library (Ignatiev, Morgado, and Marques-Silva 2018). For the KP we used the metrics estimated in Franco et al. 2020.

4.4.5 Statistical analysis

Python (version 3.6) was used to sample and solve instances. The R programming language was used to analyze the behavioral data. All of the linear mixed models (LMM), generalized logistic mixed models (GLMM) and censored linear mixed models (CLMM) included random effects on the intercept for participants (unless otherwise stated). Different models were selected according to the data structure. GLMM were used for models with binary dependent variables, LMM were used for continuous dependent variables and CLMM were used for censored continuous dependent variables (e.g., time-on-task).

All the models were fitted using a Bayesian framework implemented using the probabilistic programming language Stan via the R package 'brms' (Bürkner 2017). Default priors were used. All population-level effects of interest had uninformative priors; i.e., an improper flat prior over the reals. Intercepts had a student-t prior with 3 degrees of freedom and a scale parameter that depended on the standard deviation of the response after applying the link function. The student-t distribution was centered around the mean of the dependent variable. Sigma values, in the case of Gaussian-link models, had a half student-t prior (restricted to positive values) with 3 degrees of freedom and a scale parameter that depended on the standard deviation of the response after applying the link function. Standard deviation of the participant-level intercept parameters had a half student-t prior that was scaled in the same way as in the sigma prior.

Each of the models presented was estimated using four Markov chains. The number of iterations per chain was by default set to 2000. This parameter was adjusted to 4000 on some models to ensure convergence. Convergence was verified using the convergence diagnostic $\hat{R}$. All models presented reach an $\hat{R} \approx 1$.

Statistical tests were performed based on the 95% credible interval estimated using the highest density interval (HDI) of the posterior distributions calculated via the R package 'parameters' (Lüdecke, Ben-Shachar, and Makowski 2020). For each statistical test we report both the median ($\beta_{0.5}$) of the posterior distribution and its corresponding credible interval ($HDI_{0.95}$).

For the knapsack task, we report the statistical results from Franco et al. 2020 if available and are, here, reported as effect estimates (β) and P-Values (P). Otherwise we used the data available at the OSF (project: `https://doi.org/10.17605/OSF.IO/T2JV7`) to run statistical tests on the behavioral data. These tests were performed and reported following the same Bayesian approach used in the TSP and 3SAT analysis.

Some trials and participants were excluded due to different reasons. In the 3SAT task, two participants were omitted from the analysis given that their performance (close to 50%) differed significantly from the group. Additionally, 10 trials (from 9 participants) were omitted given that no answer was given. One participant was excluded from the time-on-task analysis since they never advanced to the response

screen before the time limit. In the TSP, one participant was excluded from the analysis given that they did not understand the instructions. This was determined during the course of the experiment. Additionally 9 trials (from 8 participants) were omitted given that no answer was selected. Finally, in the knapsack task, 13 trials (from 8 participants) were excluded in which no response was made.

4.4.6 Data and code availability

The behavioral data and the data analysis code are both available at the Open Science Framework. The 3SAT and TSP tasks are also available there (project: `https://osf.io/tekqa/`).

Acknowledgments

The authors thank Elizabeth Bowman for her support of the laboratory experiments.

Funding

This research was supported by a University of Melbourne Graduate Research Scholarship from the Faculty of Business and Economics (Franco) and a Kinsman Scholarship (Doroc). Bossaerts acknowledges financial support through a R@MAP Chair from the University of Melbourne.

Author contributions

CM, JPF, KD, PB and NY designed the study; NY and JPF performed the instance selection; KD and JPF programmed the experimental tasks; KD ran a pilot version of this study; JPF performed data collection and analysis; JPF, CM, KD, NY and PB wrote the manuscript.

Appendices

Appendix A Satisfiability and TCC

In the results section we found that TCC had a negative effect on performance. Since we are interested in understanding what generic features make instances hard for humans to solve, we explored whether TCC and satisfiability interact to make instances harder. We first explore the interaction of TCC and satisfiability on performance. Performance was not affected by satisfiability in low TCC instances on all three problems (TSP: $\beta_{0.5} = -0.16$, $HDI_{0.95} = [-0.85, 0.55]$, Table F.2 Model 4; 3SAT: $\beta_{0.5} = -0.52$, $HDI_{0.95} = [-1.27, 0.11]$, Table F.1 Model 4; KP: $\beta = -0.250$, $P = 0.355$, (Franco et al. 2020); marginal effect of satisfiability, GLMM). Moreover, in line with the previous study on the KP, we found a negative effect of TCC on both satisfiable and unsatisfiable instances for both problems considered (TSP: $\beta_{0.5}^{sat} = -2.07$, $HDI_{0.95}^{sat} = [-2.64, -1.55]$, $\beta_{0.5}^{unsat} = -2.16$, $HDI_{0.95}^{unsat} = [-2.74, -1.62]$, Table F.2 Model 4; 3SAT: $\beta_{0.5}^{sat} = -2.06$, $HDI_{0.95}^{sat} = [-2.56, -1.59]$, $\beta_{0.5}^{unsat} = -0.77$, $HDI_{0.95}^{unsat} = [-1.49, -0.13]$, Table F.1 Model 4; the effect of TCC on performance for satisfiable and unsatisfiable instances, respectively, GLMM). Interestingly, in the 3SAT problem we found that the reduction in performance due to TCC was larger for satisfiable instances ($\beta_{0.5} = -1.29$, $HDI_{0.95} = [-2.11, -0.46]$, interaction effect of TCC and satisfiability on performance, GLMM; Table F.1 Model 4). In contrast, in the TSP, as with the KP, the size of the effect of TCC on performance was similar for both satisfiable and unsatisfiable instances ($\beta_{0.5} = 0.10$, $HDI_{0.95} = [-0.65, 0.90]$, interaction effect of TCC and satisfiability on performance, GLMM; Table F.2 Model 4). This suggests that unlike the KP and TSP, in the 3SAT there is an interaction effect between TCC and satisfiability on performance, which makes satisfiable instances with high TCC harder than the rest.

When analyzing how satisfiability affected time for different levels of TCC we found different results across problems. In the TSP there was no interaction effect on time between satisfiability and TCC ($\beta_{0.5} = 0.002$, $HDI_{0.95} = [-0.054, 0.058]$, interaction effect of satisfiability and TCC on time-on-task, CLMM; Table F.4 Model 7), meaning that both properties had independent effects on time-on-task. In contrast, in 3SAT the effect of TCC was modulated by satisfiability in such a way that there was no effect of TCC when the instance was unsatisfiable ($\beta_{0.5} = 0.022$, $HDI_{0.95} = [-0.016, 0.063]$, marginal effect of TCC on time-on-task for unsatisfiable instances, CLMM; Table F.3 Model 8). In summary, we only found an interaction effect between satisfiability and TCC in the 3SAT. This was the case for both performance and time-on-task.

Appendix B Search strategies

Generic instance-level complexity metrics are able to explain differences in performance and time-spent across instances. However, it remains an open question whether the generic properties can shed light into how humans solve problems. To explore this question we investigated whether instance-level metrics could explain differences in the number of clicks across instances. This analysis was performed for TSP and 3SAT. In both tasks, participants had the opportunity to click on cities

or literals throughout the trial, whereas in the KP clicking on items was not possible. Note that while for the TSP there was no limit in the number of clicks, in the 3SAT participants were only allowed to make a maximum of 20 clicks per trial. The purpose of this limit was to discourage participants from using a trial-and-error strategy to solve the instances. Overall, the limit was reached in 11.5% of the trials.

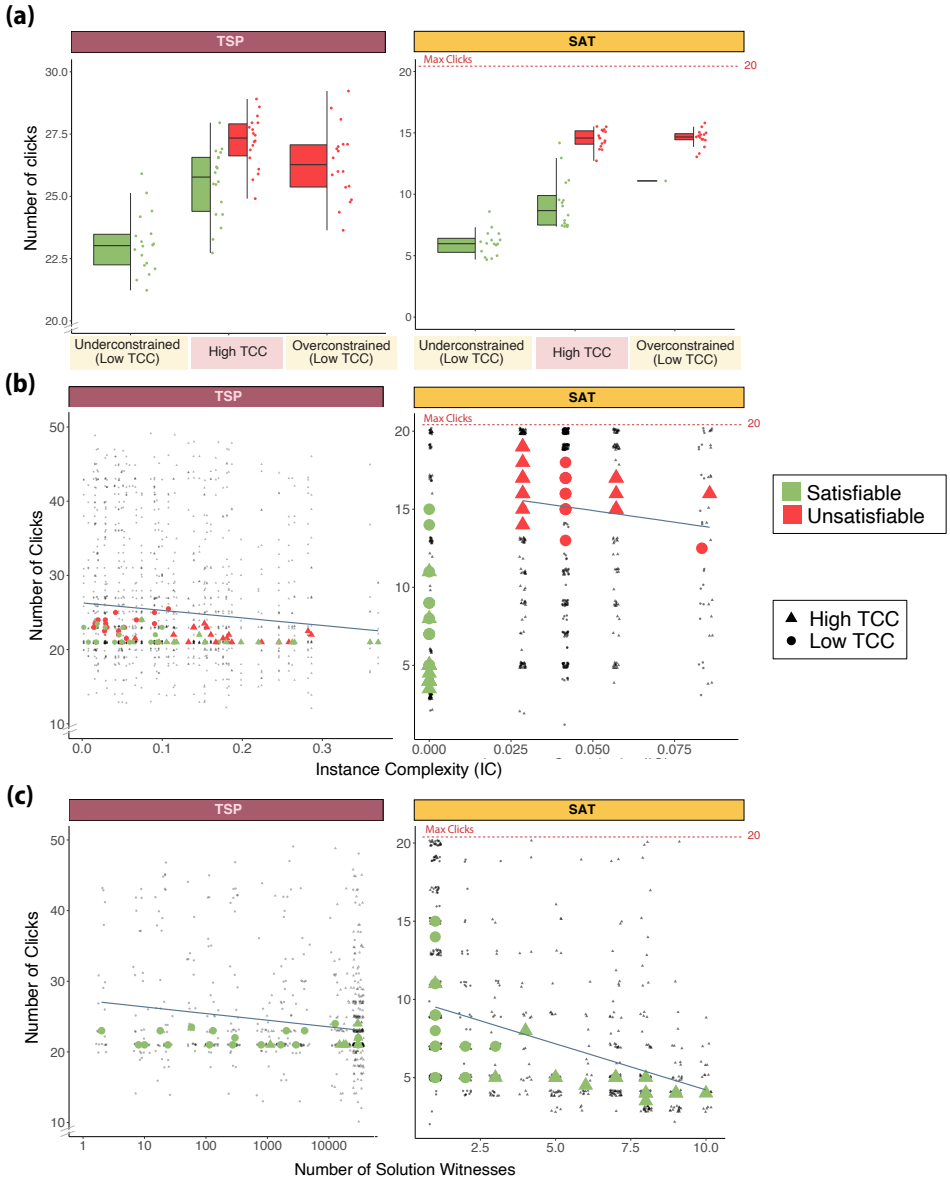
The number of clicks is a useful metric in studying the algorithms implemented by humans. Specifically, the number of clicks is related to the way that the problem state-space is explored. In the 3SAT, the state-space consists of all possible on-off switch setups (2^5 possible combinations) while in the TSP the state-space consists of all possible ordered path selections ($2^{\binom{20}{2}} = 2^{190}$ possible combinations). Arguably, participants search the state-space by clicking on different state combinations in order to decide whether an instance is satisfiable or not. Differences in the quantity of clicks used to solve an instance can shed light into how the state-space is explored (under the assumption that the state-space is explored by clicking on elements in the task).

We investigated whether generic instance-level complexity metrics could capture differences in the number of clicks. We found that participants performed more clicks on instances with high TCC, compared to low TCC, in 3SAT and TSP (TSP: $\beta_{0.5} = 1.66$, $HDI_{0.95} = [1.01, 2.33]$, GLMM Table F.8 Model 1; 3SAT: $\beta_{0.5} = 1.88$, $HDI_{0.95} = [1.23, 2.54]$, CLMM, Table F.7 Model 1; effect of TCC on number of clicks; Fig B.1c). Additionally, less clicks were performed on satisfiable instances compared to unsatisfiable ones (TSP: $\beta_{0.5} = -2.48$, $HDI_{0.95} = [-3.13, -1.85]$, LMM Table F.8 Model 2; 3SAT: $\beta_{0.5} = -7.41$, $HDI_{0.95} = [-7.95, -6.88]$, CLMM, Table F.7 Model 2; effect of satisfiability on number of clicks; Fig B.1b). We explored how these two metrics jointly affected the length of search and we found that both effects were still significant when controlling for each other in the TSP (Table F.8 Model 3; Fig B.1b). However, in 3SAT the positive effect of TCC on the number of clicks was only present on satisfiable instances (Table F.7 Model 3).

We then explored how the solution-space complexity metrics affected the length of search. Among satisfiable instances a higher number of witnesses was related to a lower amount of clicks (TSP: $\beta_{0.5} = -0.41$, $HDI_{0.95} = [-0.54, -0.26]$, LMM Table F.8 Model 5; 3SAT: $\beta_{0.5} = -0.59$, $HDI_{0.95} = [-0.69, -0.49]$, CLMM, Table F.7 Model 5; effect of number of witnesses on the number of clicks; Fig B.1c). Additionally, we found that a higher IC was related to lower number of clicks in the TSP and on unsatisfiable 3SAT instances (TSP: $\beta_{0.5} = -10.22$, $HDI_{0.95} = [-13.87, -6.37]$, LMM, Table F.8 Model 4; 3SAT: $\beta_{0.5} = -29.88$, $HDI_{0.95} = [-54.52, -6.63]$, CLMM, Table F.7 Model 4); effect of IC on number of clicks; Fig B.1b). We excluded satisfiable 3SAT instances from the analysis since we are unable to disentangle the effect of IC and satisfiability; all satisfiable instances have an $IC = 0$. In the TSP we investigated the joint effect of IC and satisfiability on the number of clicks and found that the effects were still significant when controlling for each other and that there was no interaction effect between the variables ($\beta_{0.5} = -5.95$, $HDI_{0.95} = [-13.43, 1.78]$, interaction effect between IC and satisfiability, LMM, Table F.8 Model 6).

Taken together, these findings suggest that the length of search in the state-space can be partially explained by properties of the instance, namely satisfiability and complexity. We found that there was an positive effect of TCC on search length and that the search was in general longer on unsatisfiable instances. Additionally, lower values of IC and number of witnesses were related to a longer search.

Figure B.1: **Number of Clicks. (a) Satisfiability and TCC.** Median number of clicks performed while solving an instance before submitting an answer. Each colored dot represents an instance of a problem. **(b) IC.** Each green and orange shape represent the median number of clicks for each instance of TSP and 3SAT problems. The blue lines represents the marginal effect of IC (LMM, Table F.8 Model 4, CLMM Table F.7 Model 4). Satisfiable 3SAT instances are excluded from the 3SAT model since we are unable to disentangle the effect of IC and satisfiability. Each black dot corresponds to the number of clicks by a single participant on a particular instance. The range of number of clicks presented for the TSP ($[10, 50]$) contains more than 98% of observations. **(c) Number of solution witnesses.** Each green shape represents the median time-on-task per instance. The blue line represents the marginal effect of the number of solution witnesses (LMM, Table F.8 Model 5 and CLMM, Table F.7 Model 5). *The box-plots represent the median, the interquartile range (IQR) and the whiskers extend to a maximum length of 1.5*IQR*



Appendix C Summary statistics

In this section we present summary statistics of the behavioral data for each of the tasks. These statistics exclude some observations as described in section 4.4.5.

C.1 Boolean satisfiability task

On average, participants chose the 'YES' option on 45% of trials (min = 28%, max = 61%). Performance did not vary during the course of the task ($\beta_{0.5} = 0.001$, $HDI_{0.95} = [-0.008, 0.011]$, main effect of trial number on performance, generalized logistic mixed model (GLMM); Table F.1 Model 1), suggesting that neither experience with the task nor mental fatigue affected task performance. However, time spent did vary throughout the task. As the task progressed they spent on average less time on a trial ($\beta_{0.5} = -0.005$, $HDI_{0.95} = [-0.006, -0.004]$, main effect of trial number on time-on-task —as a proportion of the maximum possible time—, censored linear mixed effects model (CLMM); Table F.3 Model 1). Overall, participants reached the maximum time allotted (110 seconds) in 16% of trials.

C.2 Traveling salesperson task

On average, participants chose the 'YES' option on 50% of trials (min = 35%, max = 60%). Consistent with our results for the 3SAT, performance did not vary during the course of the task ($\beta_{0.5} = 0.00$, $HDI_{0.95} = [-0.003, 0.013]$, main effect of trial number on performance, GLMM; Table F.2 Model 1), but participants spent less time on a trial as they progressed ($\beta_{0.5} = -0.002$, $HDI_{0.95} = [-0.003, -0.001]$, main effect of trial number on time-on-task —as a proportion of maximum possible time—), linear mixed effects model (LMM); Table F.4 Model 1). Overall, participants reached the limit of 40 seconds in 42.9% of trials.

C.3 Knapsack decision task

On average, participants chose the 'YES' option in 48.1% of trials (min = 0.32, max = 0.60, $SD = 0.06$). Performance did not vary during the course of the task ($\beta = 0.005$, $P = 0.196$, main effect of trial number on performance, GLMM; Franco et al. 2020).

Appendix D TCC and the number of witnesses

It is feasible that the effect of TCC on performance, on satisfiable instances, is driven by the number of witnesses. After all, TCC is constructed from a metric of expected constrainedness (α). We thus examined the link between these features of an instance. As expected, we found that the number of witnesses of low TCC instances was significantly higher than that of instances with high TCC in all three problems ($P_{SAT} < 0.001$, $P_{TSP} < 0.001$, $P_{KP} < 0.001$, p-values of unpaired t-tests; Fig 4.2.3). This corroborates the link between the typical-case constrainedness (α) and the solution-space constrainedness of satisfiable instances.

Based on the previous results, we hypothesized that the effect on performance of TCC (on satisfiable instances) is driven by the number of witnesses. To test this hy-

pothesis, we studied the effect of TCC on performance while controlling for the number of witnesses. In line with our conjecture, we found that once we controlled for the number of witnesses the marginal effect of TCC on performance was not significant on all three problems (3SAT: $\beta_{0.5} = 0.47$, $HDI_{0.95} = [-0.30, 1.22]$; TSP: $\beta_{0.5} = -0.12$, $HDI_{0.95} = [-0.84, 0.68]$; KP: $\beta_{0.5} = 0.17$, $HDI_{0.95} = [-0.57, 0.90]$; marginal effect of TCC on performance, GLMM; Table F.5 Models 2,5,8). We studied further this relation and tested whether there was an interaction effect of TCC and the number of witnesses on performance. The results were different across problems. We found a significant interaction in the KP, an inconclusive result in the 3SAT and a non-significant results in the TSP (KS: $\beta_{0.5} = 0.54$, $HDI_{0.95} = [0.26, 0.81]$; 3SAT: $\beta_{0.5} = 0.56$, $HDI_{0.95} = [-0.00, 1.22]$; TSP: $\beta_{0.5} = -0.26$, $HDI_{0.95} = [-0.66, 0.15]$; interaction effect between TCC and number of witnesses on performance, GLMM; Table F.5 Models 3,6,9). Taken together, these results suggest that the effect of TCC on performance is, at least partially, driven by the number of witnesses. However, on some problems, TCC might affect human performance through other mechanisms as well.

Appendix E Instance complexity in 3SAT

Unlike TSP and KP, the IC metric takes a values of zero ($IC = 0$) when the instance is satisfiable; by definition an instance is only satisfiable if the maximum number of clauses set to TRUE is equal to the number of clauses in the instance (i.e., $IC = 0$). This entails that IC would only be able to explain differences in performance on unsatisfiable instances, which (given our sampling procedure) are half of the instances used in the task. However, we did not find evidence for this explanation when we restricted our analysis to unsatisfiable instances ($R^2 = 0.001$ for unsatisfiable 3SAT instances). For this set of instances the positive relation was not significant in the 3SAT, but significant in KP and TSP (KP: $\beta_{0.5} = 13.48$, $HDI_{0.95} = [10.11, 17.30]$, Table F.6 Model 3; TSP: $\beta_{0.5} = 20.10$, $HDI_{0.95} = [15.39, 24.96]$, Table F.2 Model 8; 3SAT: $\beta_{0.5} = 9.81$, $HDI_{0.95} = [-14.59, 33.94]$, Table F.1 Model 7; the effect of IC on performance for unsatisfiable instances, GLMM). This suggests that the performance variance explained by IC in 3SAT instances might be driven by satisfiability. However, we are unable to disentangle the effect of IC and satisfiability given that all 3SAT satisfiable instance have $IC = 0$. Overall, these results indicate that IC is able to explain variance in performance across instances, but to a lesser degree in 3SAT. Moreover, the effect of IC on performance in the 3SAT might be incongruously driven by satisfiability.

Appendix F Supplementary Tables

Table F.1: **Human performance in the Boolean satisfiability task.** Logistic regressions with random intercept effects for participants relating the accuracy on an instance and trial number (1), typical-case complexity (TCC) (2), constrainedness region (3), TCC and satisfiability (4), time-on-task (5), instance complexity (IC) (6), IC on unsatisfiable instances (7) as well as satisfiability (8). *Parameter estimates correspond to the median of the posterior distribution ($\beta_{0.5}$) and the 95% HDI credible interval ($HDI_{0.95}$). ELPD denotes the expected log posterior predictive density.*

	Dependent variable: Human performance							
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Trial number	0 [-0.01,0.01]							
TCC		-1.58 [-1.95,-1.2]		-0.77 [-1.49,-0.13]				
Overconstrained			1.39 [0.93,1.86]					
Underconstrained			1.82 [1.31,2.4]					
Satisfiability				-0.52 [-1.27,0.11]				-1.35 [-1.73,-0.99]
TCC:Satisfiability				-1.29 [-2.11,-0.46]				
Time-on-task					-0.02 [-0.02,-0.01]			
IC						30.3 [21.95,39.24]	9.81 [-14.59,33.94]	
Intercept	1.95 [1.59,2.26]	2.99 [2.59,3.43]	1.41 [1.13,1.75]	3.33 [2.73,3.97]	3.15 [2.53,3.73]	1.51 [1.23,1.8]	2.99 [1.58,4.29]	2.84 [2.43,3.22]
Observations	1398	1398	1398	1398	1335	1398	675	1398
ELPD	-533.3	-493.27	-493.42	-456.8	-487.73	-505.05	-138.46	-503.36

Table F.2: **Human performance in the traveling salesperson task.** Logistic regressions with random intercept effects for participants relating the accuracy on an instance and trial number (1), typical-case complexity (TCC) (2), constrainedness region (3), TCC and satisfiability (4), time-on-task (5), satisfiability (6), instance complexity (IC) (7), as well as IC on unsatisfiable instances (8). *Parameter estimates correspond to the median of the posterior distribution ($\beta_{0.5}$) and the 95% HDI credible interval ($HDI_{0.95}$). ELPD denotes the expected log posterior predictive density.*

	Dependent variable: Human performance							
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Trial number	0 [0,0.01]							
TCC		-2.1 [-2.5,-1.73]		-2.16 [-2.74,-1.62]				
Overconstrained			2.18 [1.65,2.73]					
Underconstrained			2.05 [1.56,2.63]					
Satisfiability				-0.16 [-0.85,0.55]		-0.06 [-0.34,0.22]		
TCC:Satisfiability				0.1 [-0.65,0.9]				
Time-on-task					-0.1 [-0.13,-0.08]			
IC							21.13 [17.63,24.91]	20.1 [15.39,24.96]
Intercept	1.66 [1.43,1.94]	3.19 [2.83,3.58]	1.09 [0.91,1.3]	3.29 [2.8,3.84]	5.36 [4.39,6.34]	1.81 [1.59,2.03]	0.14 [-0.14,0.4]	0.52 [-0.21,1.29]
Observations	1575	1575	1575	1575	1575	1575	1575	787
ELPD	-656.28	-578.48	-579.52	-580.43	-612.95	-656.93	-534.35	-241.48

Table F.3: **Time-on-task in the Boolean satisfiability task.** Censored linear regressions with random intercept effects for participants relating the time spent on an instance and trial number (1), typical-case complexity (TCC) (2), constrainedness region (3), satisfiability (4), number of solution witnesses (5), instance complexity (IC) (6), IC on unsatisfiable instances (7), as well as TCC and satisfiability (8). *Parameter estimates correspond to the median of the posterior distribution ($\beta_{0.5}$) and the 95% HDI credible interval ($HDI_{0.95}$). ELPD denotes the expected log posterior predictive density.*

	Dependent variable: Time-on-task							
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Trial number	-0.01 [-0.01,0]							
TCC		0.15 [0.12,0.18]						0.02 [-0.02,0.06]
Overconstrained			0.07 [0.04,0.11]					
Underconstrained			-0.35 [-0.39,-0.32]					
Satisfiability				-0.32 [-0.35,-0.29]				-0.42 [-0.46,-0.38]
No. of witnesses					-0.04 [-0.05,-0.04]			
IC						6.04 [5.41,6.7]	-0.56 [-1.91,0.78]	
TCC:Satisfiability								0.21 [0.16,0.27]
Intercept	0.68 [0.65,0.72]	0.51 [0.4,0.62]	0.65 [0.54,0.75]	0.74 [0.64,0.84]	0.59 [0.51,0.69]	0.45 [0.35,0.57]	0.77 [0.63,0.91]	0.73 [0.62,0.83]
Observations	1335	1335	1335	1335	691	1335	644	1335
ELPD	-683.08	-456.79	-268.31	-296.9	-15.46	-340.59	-128.73	-224.94

Table F.4: **Time-on-task in the traveling salesperson task.** Censored linear regressions with random intercept effects for participants relating the time spent on an instance and trial number (1), typical-case complexity (TCC) (2), constrainedness region (3), satisfiability (4), number of solution witnesses (scaled via natural logarithm) (5), instance complexity (IC) (6), as well as TCC and satisfiability (7). *Parameter estimates correspond to the median of the posterior distribution ($\beta_{0.5}$) and the 95% HDI credible interval ($HDI_{0.95}$). ELPD denotes the expected log posterior predictive density.*

	Dependent variable: Time-on-task						
	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Trial number	0.00 [0.00,0.00]						
TCC		0.12 [0.09,0.15]					0.12 [0.08,0.16]
Overconstrained			-0.02 [-0.06,0.01]				
Underconstrained			-0.2 [-0.23,-0.16]				
Satisfiability				-0.17 [-0.2,-0.15]			-0.17 [-0.21,-0.14]
No. of witnesses (ln)					-0.02 [-0.03,-0.02]		
IC						-0.74 [-0.9,-0.58]	
TCC:Satisfiability							0.00 [-0.05,0.06]
Intercept	0.97 [0.87,1.08]	0.85 [0.75,0.95]	0.97 [0.87,1.06]	1.00 [0.89,1.1]	0.99 [0.89,1.09]	1.00 [0.9,1.11]	0.94 [0.84,1.04]
Observations	1575	1575	1575	1575	788	1575	1575
ELPD	-515.38	-499.04	-460.55	-459.22	-189.73	-491.96	-426.44

Table F.5: **Human performance an the number of solution witnesses.** Logistic regressions with random intercept effects for participants with accuracy as dependent variable. The data included on each regression is comprised of the satisfiable instances of one of the three tasks considered: 3SAT (1-3), TSP (4-6) and KP (7-9). Regressions (1), (4) and (7) include the the number of witnesses alone as regressor (the number of witnesses for the TSP is scaled via natural logarithm). Models (2), (5) and (8) include TCC, additionally, as regressor. Models (3), (5) and (8) include the interaction between TCC and number of witnesses as well. *Parameter estimates correspond to the median of the posterior distribution ($\beta_{0.5}$) and the 95% HDI credible interval ($HDI_{0.95}$). ELPD denotes the expected log posterior predictive density.*

	Dependent variable: Human performance								
	3SAT			TSP			KP		
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
No. of witnesses	0.62 [0.49,0.79]	0.7 [0.51,0.91]	0.63 [0.44,0.85]				0.26 [0.19,0.34]	0.29 [0.17,0.41]	0.11 [-0.03,0.25]
TCC		0.47 [-0.3,1.22]	-0.42 [-1.71,0.7]		-0.12 [-0.84,0.68]	2.24 [-1.5,6.13]		0.17 [-0.57,0.9]	-1.77 [-2.95,-0.44]
TCC:No. of witnesses			0.56 [0,1.22]						0.54 [0.26,0.81]
No. of witnesses (ln)				0.45 [0.37,0.53]	0.44 [0.33,0.54]	0.68 [0.25,1.03]			
TCC:No. of witnesses (ln)						-0.26 [-0.66,0.15]			
Intercept	-0.02 [-0.62,0.53]	-0.52 [-1.58,0.47]	-0.21 [-1.19,0.93]	-1.07 [-1.78,-0.46]	-0.92 [-2.16,0.21]	-3.2 [-6.83,0.73]	0.41 [-0.04,0.83]	0.22 [-0.7,1.22]	1.55 [0.35,2.78]
Observations	723	723	723	788	788	788	716	716	716
ELPD	-258.83	-259.54	-258.59	-244.8	-245.4	-245.76	-303.21	-303.92	-297.45

Table F.6: **Human performance in the knapsack task.** Logistic regressions with random intercept effects for participants relating the accuracy on an instance and satisfiability (1), instance complexity (IC) (2), and IC on only unsatisfiable instances. *Parameter estimates correspond to the median of the posterior distribution ($\beta_{0.5}$) and the 95% HDI credible interval ($HDI_{0.95}$). ELPD denotes the expected log posterior predictive density.*

	Dependent variable: Human performance		
	(1)	(2)	(3)
Satisfiability	-0.29		
	[-0.57,0.01]		
IC		9.05	13.48
		[7.2,11.02]	[10.11,17.3]
Intercept	1.79	0.59	0.47
	[1.51,2.1]	[0.28,0.92]	[0.08,0.92]
Observations	1427	1427	711
ELPD	-637.57	-574.82	-253.01

Table F.7: **Number of clicks in the Boolean satisfiability task.** Censored linear regressions with random intercept effects for participants relating the number of clicks performed on an instance and typical-case complexity (TCC) (1), satisfiability (2), TCC and satisfiability (3), instance complexity (IC) on unsatisfiable instances (4), as well as the number of solution witnesses on satisfiable instances (5). *Parameter estimates correspond to the median of the posterior distribution ($\beta_{0.5}$) and the 95% HDI credible interval ($HDI_{0.95}$). ELPD denotes the expected log posterior predictive density.*

	Dependent variable: Number of clicks				
	(1)	(2)	(3)	(4)	(5)
TCC	1.88 [1.23,2.54]		0.06 [-0.66,0.84]		
Satisfiability		-7.41 [-7.95,-6.88]	-8.8 [-9.51,-8.05]		
TCC:Satisfiability			2.91 [1.81,3.88]		
IC				-29.88 [-54.52,-6.63]	
No. of witnesses					-0.59 [-0.69,-0.49]
Intercept	10.43 [9.19,11.67]	15.15 [13.95,16.39]	15.11 [13.76,16.4]	16.4 [14.19,18.82]	10.09 [9.23,10.96]
Observations	1375	1375	1375	664	711
ELPD	-4111.67	-3825.19	-3794.12	-1645.16	-2016.33

Table F.8: **Number of clicks in the traveling salesperson task.** Linear regressions with random intercept effects for participants relating the number of clicks performed on an instance and typical-case complexity (TCC) (1), satisfiability (2), TCC and satisfiability (3), instance complexity (IC) (4), the number of solution witnesses (transformed via natural logarithm) on satisfiable instances (5), as well as IC and satisfiability (6). *Parameter estimates correspond to the median of the posterior distribution ($\beta_{0.5}$) and the 95% HDI credible interval ($HDI_{0.95}$). ELPD denotes the expected log posterior predictive density.*

	Dependent variable: Number of clicks					
	(1)	(2)	(3)	(4)	(5)	(6)
TCC	1.66 [1.01,2.33]		0.89 [0.05,1.83]			
Satisfiability		-2.48 [-3.13,-1.85]	-3.23 [-4.12,-2.3]			-1.77 [-2.81,-0.65]
TCC:Satisfiability			1.53 [0.2,2.77]			
IC				-10.22 [-13.87,-6.37]		-6.71 [-12.41,-0.87]
TCC:No. of witnesses (ln)					-0.41 [-0.54,-0.26]	
IC:Satisfiability						-5.95 [-13.43,1.78]
Intercept	24.21 [21.99,26.32]	26.38 [24.09,28.56]	26 [23.61,28.28]	26.29 [24.04,28.61]	27.31 [25.52,29.17]	27.2 [24.87,29.46]
Observations	1575	1575	1575	1575	788	1575
ELPD	-5236.1	-5220.82	-5207.65	-5234.03	-2544.04	-5207.35

References

Arora, Sanjeev. and Boaz. Barak (2009). *Computational complexity : a modern approach*. Cambridge University Press, p. 579. ISBN: 0521424267.

Ausiello, G. et al. (1999). *Complexity and Approximation : Combinatorial Optimization Problems and Their Approximability Properties*. Springer Berlin Heidelberg, p. 524. ISBN: 9783642635816.

Blum, Manuel and Santosh Vempala (2020). "The complexity of human computation via a concrete model with an application to passwords". In: *Proceedings of the National Academy of Sciences of the United States of America* 117.17, pp. 9208–9215. ISSN: 10916490. DOI: 10.1073/pnas.1801839117. URL: www.pnas.org/cgi/doi/10.1073/pnas.1801839117.

Bogdanov, Andrej and Luca Trevisan (2006). "Average-Case Complexity". In: *arXiv preprint cs/0606037*. URL: http://arxiv.org/abs/cs/0606037.

Bourgin, David et al. (2017). "The Structure of Goal Systems Predicts Human Performance". In: *Proceedigns of the 39th Annual Meeting of the Cognitive Science Society*. Ed. by G Gunzelmann et al. Austin, TX: Cognitive Science Society, pp. 1660–1665.

Budzynski, Louise, Federico Ricci-Tersenghi, and Guilhem Semerjian (Feb. 2019). "Biased landscapes for random Constraint Satisfaction Problems". In: *Journal of Statistical Mechanics: Theory and Experiment* 2019.2, p. 023302.

Bürkner, Paul-Christian (2017). "brms: An R Package for Bayesian Multilevel Models Using Stan". In: *Journal of Statistical Software* 80.1, pp. 1–28. DOI: 10.18637/jss.v080.i01.

Cheeseman, Peter, Bob Kanefsky, and William M Taylor (1991). "Where the Really Hard Problems Are". In: *The 12nd International Joint Conference on Artificial Intelligence*, pp. 331–337. ISBN: 1-55860-160-0. DOI: 10.1.1.97.3555.

De Visscher, Alice and Marie Pascale Noël (2014). "The detrimental effect of interference in multiplication facts storing: Typical development and individual differences". In: *Journal of Experimental Psychology: General* 143.6, pp. 2380–2400. ISSN: 00963445. DOI: 10.1037/xge0000029.

Dry, Matthew et al. (2006). "Human Performance on Visually Presented Traveling Salesperson Problems with Varying Numbers of Nodes". In: *The Journal of Problem Solving* 1.1. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1004. URL: http://dx.doi.org/10.7771/1932-6246.1004.

Franco, Juan Pablo et al. (2020). "Structural properties of individual instances predict human effort and performance on an NP-Hard problem". In: *bioRxiv*. DOI: 10.1101/405449. URL: https://www.biorxiv.org/content/early/2020/07/21/405449.

Frixione, Marcello (2001). "Tractable competence". In: *Minds and Machines* 11.3, pp. 379–397. ISSN: 09246495. DOI: 10.1023/A:1017503201702.

Gecode Team (2006). *Gecode: Generic Constraint Development Environment*. URL: http://www.gecode.org.

Gent, Ian P and Toby Walsh (1996). "The TSP phase transition". In: *Artificial Intelligence* 88.1-2, pp. 349–358. ISSN: 00043702. DOI: 10.1016/S0004-3702(96)00030-6.

Gershman, Samuel J, Eric J Horvitz, and Joshua B Tenenbaum (2015). "Computational rationality: A converging paradigm for intelligence in brains, minds, and

machines". In: *Science* 349.6245, pp. 273–278. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.aac6076.

Gigerenzer, Gerd and Wolfgang Gaissmaier (Jan. 2011). "Heuristic decision making". In: *Annual Review of Psychology* 62, pp. 451–482. ISSN: 00664308. DOI: 10.1146/annurev-psych-120709-145346.

Gigerenzer, Gerd. and Reinhard. Selten (2001). *Bounded rationality : the adaptive toolbox*. MIT Press, p. 377. ISBN: 9780262072144.

Griffiths, Thomas L., Falk Lieder, and Noah D. Goodman (2015). "Rational use of cognitive resources: Levels of analysis between the computational and the algorithmic". In: *Topics in Cognitive Science* 7.2, pp. 217–229. ISSN: 17568765. DOI: 10.1111/tops.12142.

Guid, Matej and Ivan Bratko (2013). "Search-Based Estimation of Problem Difficulty for Humans". In: *Artificial Intelligence in Education. AIED 2013. Lecture Notes in Computer Science*. Ed. by Lane H.C. et al. Vol. 7926. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-39112-5{_}131.

Hill, Raymond R and Charles H Reilly (2000). "Effects of coefficient correlation structure in two-dimensional knapsack problems on solution procedure performance". In: *Management Science* 46.2, pp. 302–317. ISSN: 00251909. DOI: 10.1287/mnsc.46.2.302.11930. URL: https://www.jstor.org/stable/2634765.

Hirtle, Stephen C. and Tommy Gärling (May 1992). "Heuristic rules for sequential spatial decisions". In: *Geoforum* 23.2, pp. 227–238. ISSN: 00167185. DOI: 10.1016/0016-7185(92)90019-Z.

Ignatiev, Alexey, Antonio Morgado, and Joao Marques-Silva (2018). "PySAT: A Python Toolkit for Prototyping with SAT Oracles". In: *SAT*, pp. 428–437. DOI: 10.1007/978-3-319-94144-8{_}26. URL: https://doi.org/10.1007/978-3-319-94144-8_26.

Johnson, D S, L. A. McGeoch, and E E Rothberg (1996). "Asymptotic experimental analysis for the Held-Karp traveling salesman bound". In: *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*. Vol. Part F1294, pp. 341–350. ISBN: 0898713668.

Kotovsky, K., J. R. Hayes, and H. A. Simon (Apr. 1985). "Why are some problems hard? Evidence from Tower of Hanoi". In: *Cognitive Psychology* 17.2, pp. 248–294. ISSN: 00100285. DOI: 10.1016/0010-0285(85)90009-X.

Krzakala, Florent et al. (June 2006). "Gibbs States and the Set of Solutions of Random Constraint Satisfaction Problems". In: *Proceedings of the National Academy of Sciences of the United States of America* 104.25, pp. 10318–23. ISSN: 0027-8424. DOI: 10.1073/pnas.0703685104.

Lüdecke, Daniel, Mattan S Ben-Shachar, and Dominique Makowski (2020). "Describe and understand your model's parameters". In: *CRAN*. DOI: 10.5281/zenodo.3731932. URL: https://easystats.github.io/parameters.

MacGregor, James N. and Yun Chu (2011). "Human Performance on the Traveling Salesman and Related Problems: A Review". In: *The Journal of Problem Solving* 3.2, p. 1. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1090. URL: http://dx.doi.org/10.7771/1932-6246.1090.

Mézard, M, G Parisi, and R Zecchina (2002). "Analytic and algorithmic solution of random satisfiability problems". In: *Science* 297.5582, pp. 812–815. ISSN: 00368075. DOI: 10.1126/science.1073287.

Monasson, Remi et al. (1999). "Determining computational complexity from characteristic 'phase transitions'". In: *Nature* 400.6740, pp. 133–137. ISSN: 0028-0836. DOI: 10.1038/22055.

Murawski, Carsten and Peter Bossaerts (2016). "How Humans Solve Complex Problems: The Case of the Knapsack Problem". In: *Nature (Scientific Reports)* 6.34851. ISSN: 2045-2322. DOI: 10.1038/srep34851.

Nudelman, Eugene et al. (2004). "Understanding random SAT: Beyond the clauses-to-variables ratio". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 3258, pp. 438–452. ISSN: 16113349. DOI: 10.1007/978-3-540-30201-8{_}33. URL: https://link.springer.com/chapter/10.1007/978-3-540-30201-8_33.

Ohlsson, Stellan (2012). "The Problems with Problem Solving: Reflections on the Rise, Current Status, and Possible Future of a Cognitive Research Paradigm 1". In: *The Journal of Problem Solving* 5.1. DOI: 10.7771/1932-6246.1144. URL: http://dx.doi.org/10.7771/1932-6246.1144.

Payne, John W., James R. Bettman, and Eric J. Johnson (1993). *The Adaptive Decision Maker*. DOI: 10.1017/cbo9781139173933.

Percus, Allon, Gabriel Istrate, and Cristopher Moore (2006). *Computational Complexity and Statistical Physics*. Oxford University Press, p. 384. ISBN: 9780199760565.

Preuschoff, Kerstin, Peter Bossaerts, and Steven R Quartz (2006). "Neural Differentiation of Expected Reward and Risk in Human Subcortical Structures". In: *Neuron* 51.3, pp. 381–390. ISSN: 08966273. DOI: 10.1016/j.neuron.2006.06.024.

Selman, Bart and Scott Kirkpatrick (Mar. 1996). "Critical behavior in the computational cost of satisfiability testing". In: *Artificial Intelligence* 81.1-2, pp. 273–295. ISSN: 0004-3702. DOI: 10.1016/0004-3702(95)00056-9.

Shepard, Roger N. and Jacqueline Metzler (Feb. 1971). "Mental rotation of three-dimensional objects". In: *Science* 171.3972, pp. 701–703. ISSN: 00368075. DOI: 10.1126/science.171.3972.701.

Simon, Herbert A (1990). "Invariants of human behavior". In: *Annual Review of Psychology* 41.1, pp. 1–19. ISSN: 00664308. DOI: 10.1146/annurev.psych.41.1.1. URL: www.annualreviews.org.

Smith-Miles, Kate and Leo Lopes (2012). "Measuring instance difficulty for combinatorial optimization problems". In: *Computers and Operations Research* 39.5, pp. 875–889. ISSN: 03050548. DOI: 10.1016/j.cor.2011.07.006. URL: http://dx.doi.org/10.1016/j.cor.2011.07.006.

Stazyk, Edmund H., Mark H. Ashcraft, and Mary S. Hamann (1982). "A network approach to mental multiplication". In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* 8.4, pp. 320–335. ISSN: 02787393. DOI: 10.1037/0278-7393.8.4.320.

Todd, Peter M and Gerd Gigerenzer (2012). *Ecological rationality: Intelligence in the world.* Evolution and cognition. Todd, Peter M.: Cognitive Science Program, Indiana University, 1101 E. 10th St., Bloomington, IN, US, 47405, peter.m.todd@gmail.com: Oxford University Press, pp. xviii, 590–xviii, 590. ISBN: 978-0-19-531544-8 (Hardcover). DOI: 10.1093/acprof:oso/9780195315448.001.0001.

Tsotsos, John K. (1990). "Analyzing vision at the complexity level". In: *Behavioral and Brain Sciences* 13.3, pp. 423–445. ISSN: 14691825. DOI: 10.1017/S0140525X00079577.

Tversky, Amos and Daniel Kahneman (1974). "Judgment under Uncertainty: Heuristics and Biases". In: *Science* 185.4157, pp. 1124–1131.

Unity 3D (2017). URL: https://unity3d.com/.

Van Hemert, Jano I. (2005). "Property analysis of symmetric travelling salesman problem instances acquired through evolution". In: *European Conference on Evolutionary Computation in Combinatorial Optimization*. Springer Berlin Heidelberg, pp. 122–131. DOI: 10.1007/978-3-540-31996-2{_}12. URL: https://link.springer.com/chapter/10.1007/978-3-540-31996-2_12.

Van Opheusden, Bas and Wei Ji Ma (2019). *Tasks for aligning human and machine planning*. DOI: 10.1016/j.cobeha.2019.07.002. URL: https://doi.org/10.1016/j.cobeha.2019.07.002.

Van Rooij, Iris et al. (Apr. 2019). *Cognition and Intractability*. Cambridge University Press. DOI: 10.1017/9781107358331.

Yadav, Nitin et al. (2020). "Is Hardness Inherent In Computational Problems? Performance Of Human And Digital Computers On Random Instances Of The 0-1 Knapsack Problem". In: *24th European Conference on Artificial Intelligence (ECAI 2020)*.

Zdeborová, Lenka and Marc Mézard (Dec. 2008). "Constraint satisfaction problems with isolated solutions are hard". In: *Journal of Statistical Mechanics: Theory and Experiment* 2008.12, P12004. ISSN: 1742-5468. DOI: 10.1088/1742-5468/2008/12/P12004.

Chapter 5

Neural Correlates of Computational Hardness

In this chapter I present the co-authored paper titled *"The dynamics of neural correlates of complex problem-solving"*. There we apply the framework presented in chapter 3 to the study of the neural processes that support complex problem-solving. Specifically, we perform an experiment in which human participants solve the knapsack decision task while undergoing functional MRI and we investigate the neural correlates of computational hardness.

The dynamics of neural correlates of complex problem-solving

Juan Pablo Franco, Peter Bossaerts, Carsten Murawski

Abstract

Everyday tasks involve solving a plethora of computationally complex problems. Yet, little is known on how the human brain supports complex problem-solving. This is particularly troublesome because intractable problems are fundamentally different to tractable problems, thus impeding direct extensions of previous approaches. In this paper we propose a framework to study problem-solving and cognitive demand in complex problem-solving. This framework is rooted in the notion that computational hardness of problems is an inherent characteristic of the problem and that it is associated to generic mathematical properties of instances of the problem. Here we investigate how these generic properties of instances are related to cognitive demand as well as to subjective markers of performance and reliability in problem-solving. To do this we performed an experiment in which participants solved several instances of the knapsack decision problem while undergoing ultra-high field functional magnetic resonance imaging (fMRI). We find that the neural correlates of computational hardness overlap with those associated to the multiple demand system (MDS). Importantly, our results show that these vary throughout the different stages of the task, supporting the premise that the MDS is a heterogeneous set of regions that play a dynamic and varying role at different stages during problem-solving. Of note, in line with our conjecture, we find neural markers of the reliability of a solution in the cingulo-opercular network. Our results extend the study of the neural processes associated with problem-solving by providing a framework for the study of intractable problems using a generic definition of cognitive demand. Moreover, the study of intrinsic properties of a problem put forward in this manuscript provides a way forward in the characterization of subjective beliefs of reliability in complex problems. Finally, our findings complement the investigation of cognitive control by providing a framework for the study of cognitive requirements of a task in a task-independent way.

5.1 Introduction

Humans face daily decisions that require them to solve complex problems. Many of the problems people face are known to be computationally intractable in the sense that the number of operations that need to be taken to find a solution grows quickly to levels that makes solving these problems infeasible. Everyday life examples of deceptively simple yet actually hard tasks include attention gating, task scheduling, shopping, routing, bin packing, and game play (van Rooij et al. 2019; Bossaerts and Murawski 2017). Behind the surface of these routine jobs lurk problems such as the knapsack problem, the traveling salesperson problem, the Hamiltonian circuit problem, the graph coloring problem, and the K-SAT problems (Boolean satisfiability problems).

Despite the relevance of complex problems in daily life, little is known about the neural processes that underlie how agents solve these problems. This is not to say that the neural underpinnings of problem-solving are altogether unexplored, but that the leap from the study of tractable problems to those that are considered to be complex (i.e., intractable, NP-hard) has not taken place. Indeed, a prominent approach in the study of problem-solving has analyzed the neural processes involved, by studying the neural correlates of difficulty of tractable problems (e.g., Fedorenko, Duncan, and Kanwisher 2013; Assem et al. 2020; Duncan and Owen 2000; Duncan 2010; Crittenden, Mitchell, and Duncan 2016). For instance, in working memory tasks, problem-solving is investigated by contrasting cases in which the amount of information that needs to be maintained in memory is modulated (e.g., Fedorenko, Duncan, and Kanwisher 2013; Assem et al. 2020; Duncan and Owen 2000). In general, this framework has been successful in characterizing a set of brain regions, the multiple-demand system (MDS), that generically correlates with difficulty across different tasks (Fedorenko, Duncan, and Kanwisher 2013; Assem et al. 2020; Duncan and Owen 2000; Crittenden, Mitchell, and Duncan 2016). This system is regularly considered to be composed of two networks: (1) the cingulo-opercular network (CON), consisting of the dorsal anterior cingulate cortex (dACC) and the anterior insula (AI), and (2) the the frontoparietal network (FPN), composed of the intraparietal sulcus (IPS) and specific regions from the lateral prefrontal cortex including the inferior frontal sulcus and the middle frontal gyrus (MFG) (Crittenden, Mitchell, and Duncan 2016; Sadaghiani and D'Esposito 2015; Jonathan D Power and Petersen 2013; Marek and Dosenbach 2019; Dosenbach, Fair, Miezin, et al. 2007; Dosenbach, Fair, A. L. Cohen, et al. 2008; Nomura et al. 2010; Jonathan D. Power et al. 2011; Seeley et al. 2007; Duncan 2010; Fedorenko, Duncan, and Kanwisher 2013; Crittenden, Mitchell, and Duncan 2016).

Overall, this approach provides a framework to study the generic neural processes involved in problem-solving. It allows for the study of commonalities in neural processes associated with problem-solving across different problems by implicitly defining a common currency, associated with difficulty, that can be studied across tasks provided that it reflects a real neural substrate: cognitive demand. Importantly, this procedure minimizes confounding effects by avoiding the need to artificially produce alternative and arbitrary benchmark tasks. This approach, however, has not been yet used to uncover the neural processes that support complex problem-solving.

A critical difficulty in extending this framework is that there is no evident way of defining *cognitive demand* in a generic way such that it can be studied across

tasks. A principled and generic way to characterize cognitive demand of a task is to formalize the task as a computational problem and analyze its *computational hardness*. Characterization of computational hardness has been previously studied at a behavioral level and alternative metrics of hardness have been shown to affect human behavior in a limited range of complex problems (MacGregor and Chu 2011; Hirtle and Gärling 1992; Kotovsky, Hayes, and H. A. Simon 1985; Carruthers, Masson, and Stege 2012; Bourgin et al. 2017; Shepard and Metzler 1971; Stazyk, Ashcraft, and Hamann 1982; Acuña and Parada 2010; Murawski and Bossaerts 2016; Guid and Bratko 2013; De Visscher and Noël 2014). However, most of the metrics proposed thus far are problem-specific and in many cases strategy-specific. Problem-specific approaches are troublesome because current behavioral studies have been limited to a narrow range of problems and it is not clear if or how these metrics could be extended to other problems. Strategy-specific quantification of hardness is specially problematic in complex problems because humans employ a plethora of strategies to solve these problems (e.g., MacGregor and Chu 2011; Acuña and Parada 2010; Hirtle and Gärling 1992; Ohlsson 2012; Gigerenzer and Gaissmaier 2011; Newell, Weston, and Shanks 2003; Payne, Bettman, and Johnson 1993; Siegler, Adolph, and Lemaire 1996). What would be particularly desirable is a problem-independent framework to characterize hardness, and thus cognitive demand, in complex problems. Here we propose a methodology in which, by focusing on generic intrinsic properties of the problem, it is possible to characterize neural invariants of complex problem-solving. These metrics are both problem- and strategy-independent and can be applied to a plethora of problems.

In the present paper we first examine the neural correlates of cognitive demand in a complex problem-solving task employing a metric of generic hardness that stems from computational complexity theory. Explicitly, we utilize a metric of hardness that arises from the study of random ensembles of *instances* (i.e., random cases of the problem). This field has characterized a source of considerable variation in computational hardness (for instances with the same input length) and related it to various structural properties of instances (Cheeseman, Kanefsky, and W. M. Taylor 1991; Percus, Istrate, and Moore 2006; Ian P. Gent et al. 1996; Yadav et al. 2020; Achlioptas, Naor, and Peres 2005; Selman and Kirkpatrick 1996; Krzakala et al. 2006). Critically, this source of complexity, which we call typical-case complexity (TCC), captures a source of generic cognitive demand associated with computational demands that can be modulated. This generic metric of hardness has already been shown to affect human behavior in complex (specifically NP-complete) problem-solving tasks, including the knapsack problem (Franco, Yadav, et al. 2020; Franco, Doroc, et al. 2021). Therefore, we hypothesized that TCC would serve as a problem-independent metric of cognitive demand that could be used to study the generic neural processes behind complex problem-solving.

Besides the complications that arise when quantifying cognitive demand of complex problems, the exploration of these problems entail another inherent complication. Complex problems usually require more time to solve, and thus, the neural processes that support problem-solving can no longer be modeled as a static system. The successful characterization of the neural underpinnings of complex problem-solving needs to take into account that this process ensues from a dynamic interplay of neural activity that generates strategies, modulates cognitive effort, all whilst keeping track of relevant markers of performance such as expected error (Neta,

Steven M Nelson, and Petersen 2017; Bossaerts 2018), expected rewards (Duverne and Koechlin 2017), level of uncertainty (Neta, Steven M Nelson, and Petersen 2017; Neta, Schlaggar, and Petersen 2014; Bossaerts 2018), among many other possible markers (Yoo, Hayden, and Pearson 2021; Koechlin 2016). To date, the neural invariants of this dynamic interplay during complex problem-solving has not yet been investigated.

A crucial obstacle in the study of this dynamic interplay in intractable (NP-complete) problems is the lack of a theoretical framework capable of characterizing relevant markers. What would be ideal in order to overcome this obstacle is a theory capable of delineating the intrinsic properties of the task that could give rise to these neural markers for intractable problems. Similar to how probability theory provides a framework to characterize task performance markers such as mean and variance (e.g., Preuschoff, Bossaerts, and Quartz 2006; D'Acremont and Bossaerts 2016; D'Acremont and Bossaerts 2008; d'Acremont, Schultz, and Bossaerts 2013; Christopoulos et al. 2009; O'Neill and Schultz 2013). The methodology put forward in this study delineates a set of intrinsic properties of NP-complete problems that could serve this purpose. Among these properties we investigated two which we hypothesized would be related to markers of expected performance and to the reliability of candidate solutions. Here we define *reliability* as the subjective belief of the degree to which a candidate solution can be depended on to be accurate. Firstly, we studied TCC, which has been shown to affect performance, and thus, is potentially linked to markers of expected performance and effort efficacy. Secondly, we considered markers related to satisfiability, an intrinsic property of NP-complete problems. Satisfiability is a property of *decision problems* (problems whose solution is either 'yes' or 'no') that represents the solution of the problem. In NP-complete problems, satisfiability captures an asymmetry that does not occur in easy (tractable) problems. Explicitly, in order to conclude that an instance is satisfiable (solution is 'yes') it suffices to find a witness (example) that satisfies all of the constraints, this solution can then be verified quickly. In contrast, to confirm that an instance is unsatisfiable (solution is 'no') requires generating a proof of non-existence. We conjectured that this asymmetry between satisfiable and unsatisfiable instances would be closely related to subjective markers of reliability. Much like variance is reflected in subjective beliefs of uncertainty in probabilistic tasks. As such, we expected to see neural markers of satisfiability in regions that have been previously shown to encode uncertainty, specifically in CON (Neta, Steven M Nelson, and Petersen 2017; Neta, Schlaggar, and Petersen 2014; Bossaerts 2018; Fouragnan, Retzler, and Philiastides 2018).

Here we investigate the neural correlates of TCC and satisfiability in the knapsack decision problem and explore the related neural dynamics. To this end, we performed a study in which participants solved several instances of the knapsack decision problem while undergoing functional magnetic resonance imaging (fMRI). Instances varied in their computational complexity (specifically TCC) and in their satisfiability. Critically, in order to be able to investigate the temporal dynamics of problem-solving at a more granular level, we employed an ultra-high field scanner for this study. This allowed us to increase both the temporal and spatial resolution of the neuroimaging data collected.

We first considered the neural correlates of TCC. We expected regions associated with the MDS to show differential brain activity between instances with high and

low TCC. We indeed found that the neural correlates of complexity overlapped with those characterized in the MDS: CON and FPN. However, the correlates varied across time, presenting a different picture at different stages in the task. Importantly, we did not find neural correlates of TCC early in the task, suggesting that the effects of complexity in calculation and control allocation are only realized later on in the trial.

Additionally, we explored the neural correlates of satisfiability. We expected the asymmetry between satisfiable and unsatisfiable instances to be reflected in a different strategy use throughout the solving stage and to reflect differences in control signals in areas that have been shown to track uncertainty. In particular, we expected to see higher activation in CON on unsatisfiable instances during late stages of the trial (Neta, Steven M Nelson, and Petersen 2017; Neta, Schlaggar, and Petersen 2014; Bossaerts 2018; Fouragnan, Retzler, and Philiastides 2018). Our findings support this hypothesis. However, we expected to see neural correlates of satisfiability only late in the trial, given that the estimation of satisfiability corresponds to solving the problem. Interestingly, and contrary to our expectations, we found significant differences in activation from early in the trial in several regions.

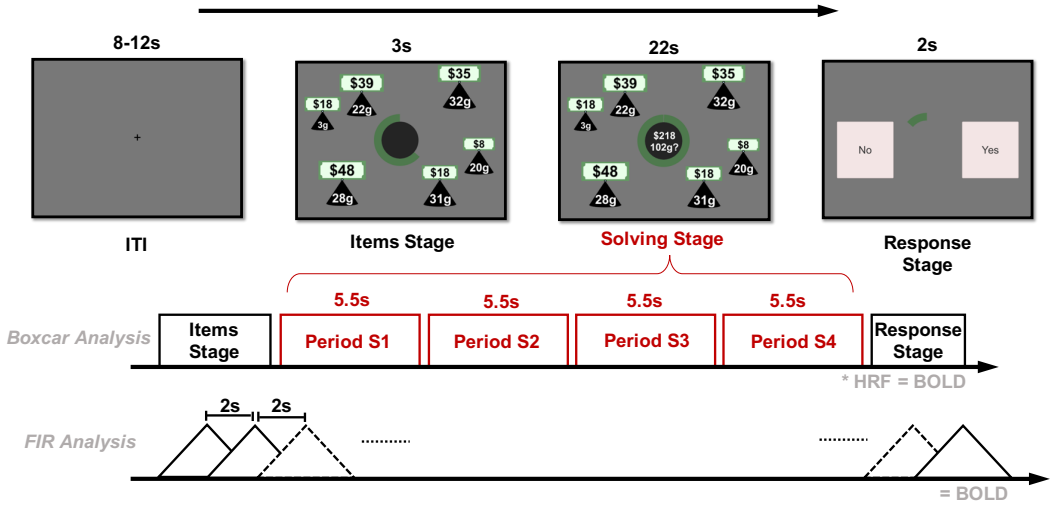
Finally, we investigated how the functional connectivity between regions of interest in the MDS changed during the solving stage of the task. We expected regions in CON, which are usually associated with proactive control allocation (Shenhav, Botvinick, and J. D. Cohen 2013; Dosenbach, Visscher, et al. 2006; Silvetti et al. 2018; Vassena, Holroyd, and Alexander 2017; Holroyd and Yeung 2012; Alexander and Brown 2011; Sestieri et al. 2014; Aben et al. 2020; Crottaz-Herbette and Menon 2006), to have higher effective connectivity on a region strongly associated with processing in mathematical problem-solving (IPS) (Matejko and Ansari 2018; Grabner et al. 2009; De Smedt, Holloway, and Ansari 2011; Arsalidou and M. J. Taylor 2011; Brannon 2006). We also expected this connectivity to be modulated by the complexity of the instance. We found, in line with our hypothesis, that during the solving stage of the task the dACC had higher connectivity to the IPS. However, contrary to our expectations, we did not find increased connectivity between AI and IPS. Moreover, we found no significant effect of TCC nor satisfiability on the strength of this connectivity.

5.2 Results

Twenty participants participated in this study. Each participant was asked to solve 56 instances of the knapsack decision task while undergoing an ultra-high field MRI brain scan. In this task participants are asked to determine whether there exists a subset of items with predefined values and weights that exceed a minimum total value while not exceeding a maximum total weight (Fig 5.2.1). Instances varied in their computational hardness (TCC) and in their satisfiability (2×2 balanced factorial design; see Materials and Methods).

Additionally, participants performed, outside the scanner, a set of complementary tasks, including the knapsack optimization task and a set of cognitive function tasks. In this section we report the behavioral results of the knapsack decision task, while the behavioral results from the complementary tasks are reported in Appendix A.

Figure 5.2.1: **Knapsack decision task. (a) Paradigm description.** The task was composed of three main stages: items stage (3s), solving stage (22s) and response stage (2s). Initially, participants were presented with a set of items of different values and weights. The green circle at the center of the screen indicated the time remaining in this stage of the trial. This stage lasted 3 seconds. Then, both capacity constraint and target profit were shown at the center of the screen. Participants decided whether there existed a subset of items for which (1) the sum of weights is lower or equal to the capacity constraint and (2) the sum of values yields at least the target profit. This stage lasted 22 seconds. Finally, participants had 2 seconds to make either a 'YES' or 'NO' response using the keyboard. A fixation cross was shown during the inter-trial interval (jittered between 8 and 12 seconds). **(b) Analysis of task stages.** The imaging data was modeled using two different approaches. We first employed a Boxcar analysis, in which the BOLD signal was modeled by convolving a canonical HRF with Boxcar functions. The solving stage was partitioned into four Boxcar response functions (periods S1-S4), and the items and response stage were each modeled with a single Boxcar function for the duration of the stage. The second type of analysis was a Finite Impulse Response (FIR) analysis. In this approach we modeled the BOLD signal directly without convolving an HRF. The entire duration of the trial was modeled employing a set of 17 tent response functions equally spaced every 2 seconds. These tent-parameters are aligned to the BOLD signal.



5.2.1 Behavioral results

Summary statistics

On average, participants chose the 'YES' option 50% of the trials (min = 25%, max = 68%). Mean *human performance*, measured as the proportion of trials in which a correct response was made, was 0.78 (min = 0.48, max = 0.95, $SD = 0.14$). Performance had a non-significant improvement as the task progressed ($\beta_{0.5} = 0.009$, $HDI_{0.95} = [-0.001, 0.021]$, main effect of trial number on performance, generalized logistic mixed model (GLMM); Table C.1 Model 1). This marginal improvement in the task performance might seem to contradict previous results which suggest that neither experience with the task nor mental fatigue affected task performance

(Franco, Yadav, et al. 2020). However, unlike Franco, Yadav, et al. 2020, we performed the task while participants underwent scanning, thus this discrepancy could be due to acclimatization to the scanner.

Accuracy and instance properties

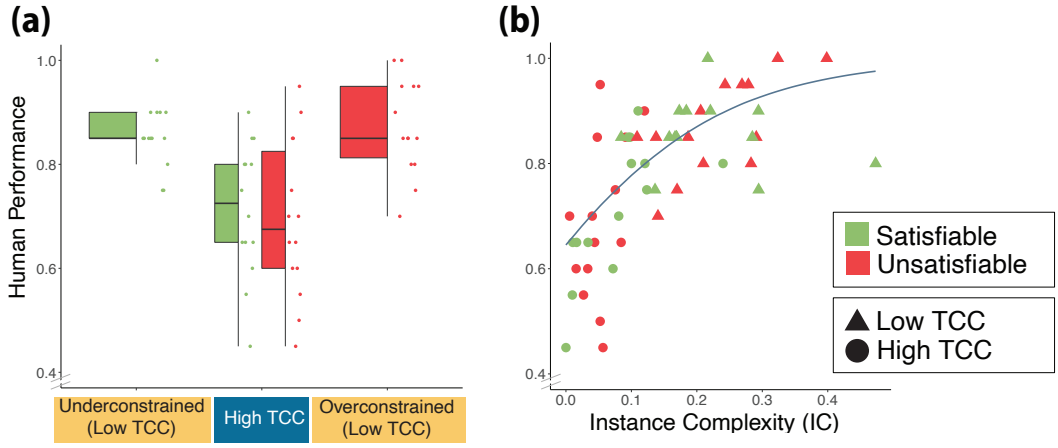
We first studied the effect of TCC on human performance. This measure is based on a prominent framework in computer science that investigates the drivers of computational hardness in computational problems by studying the difficulty of randomly generated instances of those problems. In the knapsack problem, TCC is explicitly connected to a set of parameters $\bar{\alpha} = (\alpha_c, \alpha_p)$ that capture the constrainedness of the problem: $\alpha_p = p/\sum_{i=1}^{N} v_i$ and $\alpha_c = c/\sum_{i=1}^{N} w_i$ (Yadav et al. 2020; Franco, Yadav, et al. 2020). These parameters determine the likelihood that a random instance is *satisfiable*; that is, that the solution is 'yes'. Specifically, they characterize where typical instances are generally satisfiable (under-constrained region), where they are unsatisfiable (over-constrained region) and where the probability of satisfiability is close to 50% (satisfiability threshold α_s). TCC is defined based on the distance of α_p to the satisfiability threshold α_s. Specifically, instances with values of α_p near the satisfiability threshold have a high typical-case complexity (*high TCC*) whereas instances further away from it—that is, in the under-constrained and over-constrained regions— have low typical-case complexity (*low TCC*).

We hypothesized that participants would have a better performance on instances with low TCC compared to those with high TCC; in line with the results by Franco, Yadav, et al. 2020. As predicted, we found that TCC had a negative effect on performance ($\beta_{0.5} = -1.10$, $HDI_{0.95} = [-1.44, -0.79]$, main effect of TCC on performance, GLMM; Table C.1 Model 2).

Another relevant feature of instances of decision problems, is their *satisfiability*. Despite the asymmetry encoded by this property, previous results suggest that there is no effect of satisfiability on human performance in the knapsack decision task (Franco, Yadav, et al. 2020). Our findings findings replicate these results ($\beta_{0.5} = 0.02$, $HDI_{0.95} = [-0.30, 0.30]$, main effect of satisfiability on performance, GLMM; Table C.1 Model 5). Moreover, we tested whether there was an interaction effect between TCC an satisfiability on performance and found no significant interaction effect ($\beta_{0.5} = 0.26$, $HDI_{0.95} = [-0.37, 0.90]$, interaction effect of TCC and satisfiability, GLMM; Table C.1 Model 6).

An additional aim of this study was to reproduce the key findings presented by Franco, Yadav, et al. 2020. We have already shown that the results regarding TCC and satisfiability are mirrored by our data and statistical analyses. Two other key findings in their study were related to two related *solution-space* metrics of complexity: Instance complexity (IC) and number of solution witnesses (i.e., the number subsets of items that satisfy both profit and capacity constraints). In order to estimate these metrics, unlike TCC, the problem needs to be solved. Concretely, harder versions of the problem require solving. For IC to be estimated, the optimization variant of the knapsack problem needs to be solved, while for the number of witnesses all of the possible sets of items that satisfy the constraints need to be found. This makes estimating these metrics more computationally intensive than estimation of TCC. Despite this drawback, these metrics capture the hardness of a single instance of the problem and therefore are more precise when predicting per-

Figure 5.2.2: **Human performance in the knapsack decision task. (a) TCC.** Each dot represents an instance; human performance corresponds the proportion of participants that solved the instance correctly. Instances are categorized according to their constrainedness region (α) and their TCC. In the underconstrained region (low TCC) the satisfiability probability is close to one, while in the overconstrained region (low TCC) the probability is close to zero. The region with a high TCC corresponds to a region in which the probability is close to 0.5. Additionally, instances are categorized according to their solution (satisfiability) which is represented by their color. *The box-plots represent the median, the interquartile range (IQR) and the whiskers extend to a maximum length of 1.5*IQR* **(b) IC.** Mean accuracy per instance and the marginal effect of IC on human performance (GLMM; Table C.1 Model 3). Higher IC is related to lower computational hardness. Instances are categorized by their TCC (shape) and satisfiability (color).



formance for each instance compared to TCC, which captures the average hardness of an ensemble of random instances.

Franco, Yadav, et al. 2020 showed that human performance was affected by both IC and the number of witnesses. Here we reproduced these findings. We found that higher values of IC were related to higher accuracy ($\beta_{0.5} = 6.54$, $HDI_{0.95} = [4.67, 8.29]$, main effect of IC, GLMM; Table C.1 Model 3). Similarly, among satisfiable instances, we found that a higher number of witnesses was related to better performance ($\beta_{0.5} = 0.20$, $HDI_{0.95} = [0.12, 0.28]$, main effect of number of witnesses in satisfiable instances, GLMM; Table C.1 Model 4). It is worth noting that the number of witnesses can only explain variability among satisfiable instances since all unsatisfiable instances have 0 witnesses.

Overall, we find a significant effect on performance of TCC, IC and number of witnesses. In contrast, we found no effect of satisfiability on performance. These results replicate previous findings (Franco, Yadav, et al. 2020) and validate that the experimentally modulated variable (TCC) successfully varied the hardness of the task.

Finally, we investigated human performance in a set of related tasks. We studied the effect of computational hardness in the knapsack optimization task and replicated previous results (Franco, Yadav, et al. 2020). Explicitly, we found that a previously proposed extension of TCC to optimization problems does, indeed, have an effect on both human performance and time-on-task (See Appendix A.1). More-

over, we explored the relation between performance in the knapsack tasks and core cognitive abilities, including working memory, episodic memory, strategy use, as well as mental arithmetic (see Appendix A.2). For this analysis, we utilized the joined data set from this study together with the data collected by Franco, Yadav, et al. 2020. Our results suggest a weak relation between these cognitive abilities and performance in the knapsack tasks. The only significant correlation (at $\alpha = 0.05$) shows a link between mental arithmetic ability and performance in the knapsack optimization task.

5.2.2 Imaging results

Whole brain analysis

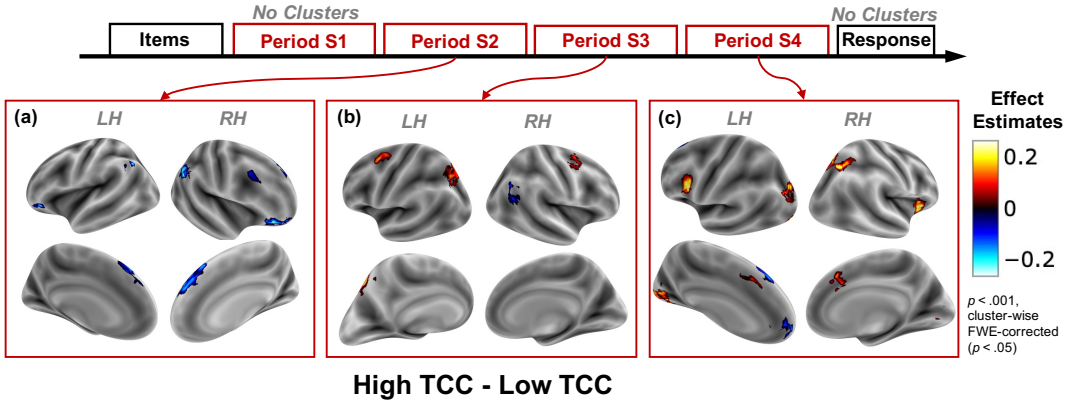
We explored the whole-brain neural correlates of two intrinsic generic properties of the problem at hand: TCC and satisfiability. Additionally, we investigated the neural correlates of response accuracy. We did this by fitting GLMs that partitioned the solving stage into four separate periods (5.5s) with an additional response stage modeled in the analysis (2s) (see Fig 5.2.1). Here we present the results of these regressions.

Neural correlates of TCC We first explored the neural correlates of TCC. We expected to see the $highTCC - lowTCC$ contrast capture differences in BOLD activation in regions previously correlated with cognitive demand (i.e., MDS). We explicitly expected to find evidence for the encoding of TCC in CON from early on during the solving stage due to its link to expected performance and reliability. Higher TCC entails, on average, lower performance and lower reliability of finding the solution (Fig 5.2.2). Note that the estimation of TCC early on in the trial is feasible, because constrainedness (and thus TCC) can be potentially estimated by performing a sum and division operations ($\alpha_p = p/\sum_{i=1}^{N} v_i$ and $\alpha_c = c/\sum_{i=1}^{N} w_i$).

We found that the neural correlates of TCC varied throughout the duration of the solving stage. We report these results in Figure 5.2.3 and present the corresponding cluster information in Table 5.2.1. Contrary to our expectations we did not find significant correlations of TCC during the first period of the solving stage. Interestingly, during the second period we did find a set of clusters that showed higher BOLD activity on instances with low TCC. These regions include the angular gyrus (AG) bilaterally, the SFG, the right MFG as well as regions in the orbitofrontal cortex (bilaterally). It is worth noting that the negative pattern found in this period might stem from a different slope in the increased task-related activation and not from differences in the sustained level of activity (see Fig 5.2.6). This pattern would align with previous results that support that FPN regions encode evidence accumulation towards a particular decision (Ploran et al. 2011; C. Gratton et al. 2017). Indeed, in the knapsack task we would expect that lower TCC would be associated with faster evidence accumulation towards a solution.

During the third period of the solving stage the TCC contrast still showed significant clusters along FPN, but the pattern overall changed, with respect to period 2, except for the cluster in the right AG. Critically, we found that a different set of regions within FPN now showed a positive correlation with TCC. Specifically, we found clusters in the left SFG, left IPS, the cerebellum, as well as a cluster in the

Figure 5.2.3: **Neural correlates of TCC.** Brain activation effect estimates (β) for the high vs. low TCC contrast ($\beta_{highTCC} - \beta_{lowTCC}$). A positive contrast represents a higher BOLD activity on instances with high TCC compared to low TCC. Significant cluster-wise FWE-corrected ($p < 0.05$) clusters (with an uncorrected threshold of $p < 0.001$) are presented for each of the contrasts estimated using the Boxcar analysis. Each panel represents a different period in the solving stage. **(a)** Period S2, **(b)** period S3, **(c)** period S4. No significant clusters were found in period S1 nor in the response stage.



right dorsolateral prefrontal cortex (dlPFC) in between the MFG and the SFG. Interestingly, the right AG kept on displaying a negative correlation with TCC during this period.

Finally, during the fourth, and last, period of the solving stage, a new set of clusters was identified. Markedly, this new set of clusters include regions from both CON, FPN, as well as significant clusters in the occipital lobe. In general, the activation in these clusters correlate positively with TCC. These include the dACC and right AI from CON as well as the precentral gyrus and the IPS from the FPN. The right IPS activation is segregated into two clusters, one medial and superior that overlaps with the precuneus and one more lateral that overlaps with the AG. The only two clusters that correlated negatively with TCC in this period are those located in the ACC, as well as a cluster in the left SFG that overlaps with the SFG cluster found in the second period.

We did not find any significant clusters during the response stage.

Neural correlates of satisfiability Understanding the processes that support problem-solving is particularly difficult for complex problems because there are many possible strategies that agents might use to solve a problem. In order to better understand these supporting processes we propose that the intrinsic properties of the problem can be employed as markers to capture invariants of these processes. One such property of decision problems, is its satisfiability. This characteristic is relevant because it encodes an asymmetry of NP-complete problems that might be related to the implementation of different strategies. Explicitly, in order to conclude that an instance is satisfiable it suffices to find a witness that satisfies all of the constraints. In contrast, to derive that an instance is unsatisfiable requires a proof

Table 5.2.1: **TCC clusters.** Significant cluster-wise FWE-corrected ($p < 0.05$) clusters (using an uncorrected threshold of $p < 0.001$) from the *High TCC - low TCC* contrast. Coordinates are in MNI space.

			Cluster statistics			Peak statistics			
Stage	Region	Side	Volume(mm^3)	β_{mean}	SEM	β_{peak}	x	y	z
S2	SFG	RH/LH	4763.6	-0.17	0.001	-0.32	13	34	61
	Orbitofrontal cortex	RH	3878.9	-0.21	0.002	-0.37	51	44	-19
	AG	RH	2662.4	-0.20	0.002	-0.31	51	-55	37
	AG	LH	897.0	-0.21	0.002	-0.29	-58	-66	36
	Orbitofrontal cortex	LH	749.6	-0.20	0.003	-0.31	-51	36	-19
	MFG	RH	495.6	-0.16	0.002	-0.20	48	17	36
S3	IPS	LH	2043.9	0.16	0.002	0.35	-11	-79	52
	Cerebelum	RH	938.0	0.11	0.002	0.19	0	-60	-25
	SFG	LH	786.4	0.14	0.002	0.19	-26	-2	52
	AG	RH	495.6	-0.17	0.002	-0.24	56	-60	36
	MFG/SFG	RH	483.3	0.12	0.002	0.17	30	-1	61
S4	Occipital Pole	LH	2732.0	0.20	0.001	0.32	-10	-97	-8
	Fusiform gyrus	LH	1888.3	0.19	0.002	0.31	-24	-79	-14
	Middle occipital gyrus	LH	1503.2	0.20	0.002	0.28	-29	-78	21
	AI	RH	1265.7	0.21	0.003	0.30	32	28	0
	Precentral gyrus	LH	1163.3	0.22	0.003	0.33	-43	4	24
	IPS (precuneus)	RH	1044.5	0.21	0.003	0.33	13	-76	60
	SFG	LH	1024.0	-0.15	0.003	-0.27	-16	36	58
	dACC	LH/RH	901.1	0.18	0.002	0.24	-2	22	40
	IPS (AG)	RH	696.3	0.24	0.004	0.35	32	-65	47
	Occipital pole	LH	667.6	0.26	0.003	0.34	-38	-95	-6
	ACC	LH	475.1	-0.22	0.003	-0.29	-5	57	8

of non-existence.[1]

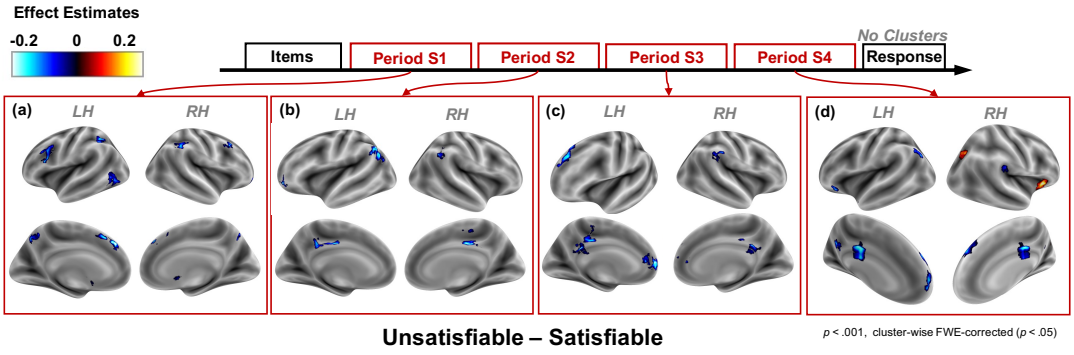
We expected the asymmetry between satisfiable and unsatisfiable instances to not only be reflected in a different strategy use throughout the solving stage, but also reflect differences in neural markers associated with reliability. Specifically, we hypothesized that satisfiable instances would be associated with higher reliability, given that once a solution witness is found, verifying that the proposed solution is correct is straightforward (a polynomial-time operation). In contrast, for unsatisfiable instances verifying a proof of non-existence might be more convoluted. Therefore, we expected regions that have been linked to monitoring of uncertainty to be more active during the trial on unsatisfiable instances compared to satisfiable ones. In particular, we conjectured higher activation of the CON, on unsatisfiable instances, during late stages of the trial (Neta, Steven M Nelson, and Petersen 2017; Neta, Schlaggar, and Petersen 2014; Bossaerts 2018; Fouragnan, Retzler, and Philiastides 2018). We expected to find neural correlates of satisfiability only late during the solving stage, given that the estimation of satisfiability corresponds to solving the problem.

We report the neural correlates of satisfiability ($Unsatisfiable - Satisfiable$) in Figure 5.2.4 and present the corresponding cluster information in Table 5.2.2. Interestingly, and contrary to our expectations, we found significant clusters from the first period of the solving stage. Moreover, significant clusters did not extend to the response screen, which was also in opposition to our hypothesis. Most of the clusters during the solving stage showed a lower BOLD activity for unsatisfiable instances. These clusters extended from period one to period four of the solving stage. Notably, the posterior cingulate showed a lower sustained activation on unsatisfiable instances

[1]Conceptually, this asymmetry reflects the conjectured null intersection between complexity classes NP-complete and co-NP-complete.

throughout the solving stage (periods S2, S3 and S4). Similarly, different clusters in the SFG had significant clusters throughout the solving stage. Additionally, similar to the clusters found for the TCC contrast, the AG showed bilateral activation during the second period of the solving stage. Interestingly, a bigger AG cluster was found on the left hemisphere compared to the right, in contrast to the right laterality predominance of AG found in the TCC contrast.

Figure 5.2.4: **Neural correlates of satisfiability.** Brain activation effect estimates (β) for the unsatisfiable vs. satisfiable contrast ($\beta_{unsatisfiable} - \beta_{satisfiable}$). A positive contrast represents a higher BOLD activity on unsatisfiable instances. Significant cluster-wise FWE-corrected ($p < 0.05$) clusters (with an uncorrected threshold of $p < 0.001$) are presented for each of the contrasts estimated using the Boxcar analysis. Each panel represents a different period in the solving stage. **(a)** Period S1, **(b)** period S2, **(c)** period S3 and , **(d)** period S4. No significant clusters were found in the response stage.



The two clusters that showed a higher activity on unsatisfiable instances were the right AI and the occipital superior cortex, both present only during period four of the solving stage. The significant cluster found in the AI is in line with our hypothesis that unsatisfiable instances are related to higher markers of uncertainty. A signal which we expected to find in CON. However, in disagreement with our hypothesis, we did not find a significant satisfiability cluster in the the dACC. This, however, coincides with alternative views that suggest a dissociation between the role of AI and dACC, where the AI is involved in monitoring of control signals or alerting, whilst the dACC is associated with task switching and active control allocation (Han, Eaton, and Marois 2019; Billeke et al. 2020).

Neural correlates of accuracy It has been hypothesized that FPN as well as CON regions encode task signals related to error detection and error expectation (Neta, Steven M Nelson, and Petersen 2017; Neta, Schlaggar, and Petersen 2014; Dosenbach, Visscher, et al. 2006). Although participants did not receive any feedback during the task, we expected to see error related signals during later stages of the trial. Although these signals would not represent the integration of novel exogenous information (since there was no feedback) we conjectured that participants would represent a subjective belief on the expected accuracy (or reward) of their answer (e.g, Duverne and Koechlin 2017).

We found only one significant cluster during the solving stage (in period one) (Fig 5.2.5a; Table 5.2.3). The other significant clusters were identified during the

Table 5.2.2: **Satisfiability clusters.** Significant cluster-wise FWE-corrected ($p < 0.05$) clusters (using an uncorrected threshold of $p < 0.001$) from the *Unsatisfiable-Satisfiable* contrast. Coordinates are in MNI space.

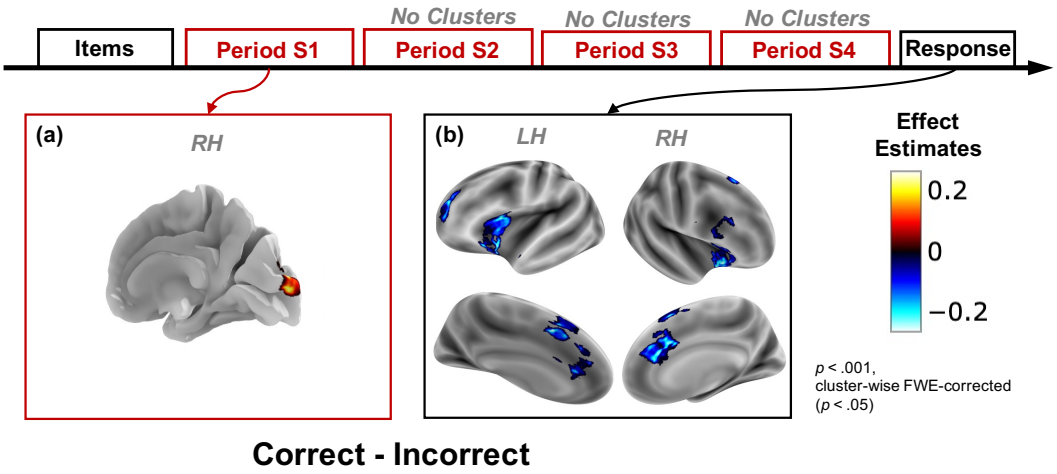
			Cluster statistics			Peak statistics			
Stage	Region	Side	Volume(mm^3)	β_{mean}	SEM	β_{peak}	x	y	z
S1	SFG	LH	1876.0	-0.14	0.002	-0.23	-3	38	42
	Supramarginal gyrus	RH	1740.8	-0.13	0.002	-0.25	54	-46	56
	Supramarginal gyrus	LH	1425.4	-0.12	0.001	-0.17	-42	-47	40
	Inferior occipital cortex	LH	1159.2	-0.13	0.002	-0.22	-61	-65	-12
	MFG	LH	905.2	-0.15	0.001	-0.20	-48	18	28
	Caudate	LH	880.6	-0.17	0.002	-0.24	-8	2	-1
	MFG	RH	868.4	-0.12	0.003	-0.23	37	30	53
	Cerebellum	LH	667.6	-0.12	0.002	-0.18	-38	-79	-49
	Precuneus	LH	516.1	-0.19	0.003	-0.26	-2	-65	44
	Frontal pole	RH	450.6	-0.14	0.002	-0.18	29	58	-9
	Caudate	RH	450.6	-0.17	0.003	-0.23	8	4	0
S2	AG	LH	2523.1	-0.19	0.002	-0.29	-62	-60	29
	Posterior cingulate	RH	696.3	-0.20	0.003	-0.25	2	-25	40
	AG	RH	585.7	-0.18	0.003	-0.28	62	-57	34
	SFG	LH	577.5	-0.23	0.004	-0.40	-38	62	-8
	Posterior cingulate	LH	479.2	-0.16	0.002	-0.19	-10	-41	37
S3	SFG	LH	1511.4	-0.15	0.002	-0.22	-21	36	55
	Anterior cingulate	LH	708.6	-0.21	0.003	-0.30	-5	52	4
	Supramarginal gyrus	RH	696.3	-0.14	0.002	-0.20	64	-28	39
	Frontal pole	RH	692.2	-0.14	0.002	-0.21	13	58	31
	Anterior cingulate	LH	593.9	-0.15	0.002	-0.24	-6	46	12
	Posterior cingulate	LH	577.5	-0.17	0.002	-0.22	-2	-28	45
	Posterior cingulate	LH	487.4	-0.18	0.004	-0.28	-2	-44	28
S4	Posterior cingulate	RH	1384.5	-0.22	0.003	-0.33	0	-18	34
	SFG	RH	1306.6	-0.18	0.002	-0.26	14	50	42
	AI	RH	901.1	0.25	0.003	0.36	32	28	0
	AG	LH	659.5	-0.24	0.003	-0.30	-48	-68	44
	SFG	LH	647.2	-0.17	0.004	-0.26	-14	52	40
	Precuneus	LH	581.6	-0.19	0.005	-0.27	-5	-57	31
	Occipital superior cortex	RH	544.8	0.17	0.005	0.27	29	-63	36
	SFG / Frontal pole	LH	512.0	-0.24	0.003	-0.34	-3	65	16
	Orbitofrontal cortex	LH	454.7	-0.23	0.003	-0.33	-46	28	-20
	Supramarginal	RH	438.3	-0.12	0.002	-0.18	54	-33	32

response stage (Fig 5.2.5b; Table 5.2.3). In line with out hypothesis, we found that activity in both FPN and CON was positively correlated with erring. Specifically, a higher activity was found for incorrect trials in the AI (bilaterally), dACC, left MFG and the right inferior frontal gyrus. In addition to the regions commonly associated with the MDS, we also found significant activation in the SFG (bilaterally), ACC and paracingulate gyrus.

ROI dynamics

Three ROIs were selected (see section 5.4.9) to investigate more closely the neural dynamics of complex problem-solving and the interplay of this process with monitoring of control signals and proactive allocation of control. We included in our analysis the dACC due to its proposed involvement in the allocation of control (Shenhav, Botvinick, and J. D. Cohen 2013; Dosenbach, Visscher, et al. 2006; Silvetti et al. 2018; Vassena, Holroyd, and Alexander 2017; Holroyd and Yeung 2012; Alexander and Brown 2011), as well as the right AI because of its involvement in encoding control signals and uncertainty in particular (Neta, Steven M Nelson, and Petersen 2017; Neta, Schlaggar, and Petersen 2014; Bossaerts 2018; Fouragnan, Retzler, and Philiastides 2018). Additionally, we selected a processing unit associated

Figure 5.2.5: **Neural correlates of accuracy.** Brain activation effect estimates (β) for the correct vs. incorrect contrast ($\beta_{correct} - \beta_{incorrect}$). A positive contrast represents a higher BOLD activity on instances that were answered correctly. Significant cluster-wise FWE-corrected ($p < 0.05$) clusters (with an uncorrected threshold of $p < 0.001$) are presented for each of the contrasts estimated using the Boxcar analysis. Each panel represents a different period in the trial. **(a)** Period S1, **(b)** response stage. No significant clusters were found in the contrasts during periods S2-S4 of the solving stage.



with mathematical calculations, the right IPS (Matejko and Ansari 2018; Brannon 2006; Arsalidou and M. J. Taylor 2011).

The BOLD activation associated with each of the ROIs is presented in Figure 5.2.6. The effect estimates β_{FIR} correspond to 2×2 FIR analysis performed for the factors TCC and satisfiability (see section 5.4.9 for details). We found a similar patterns for AI and dACC. Overall, in both ROIs the activity rose throughout the task and quickly decreased around the time the solving stage ended. The activity pattern in the IPS showed a different pattern to that of CON regions. In this region, the activity increased quickly early on in the trial and was sustained until it started decreasing later on. The moment at which the decrease started was modulated by TCC and satisfiability.

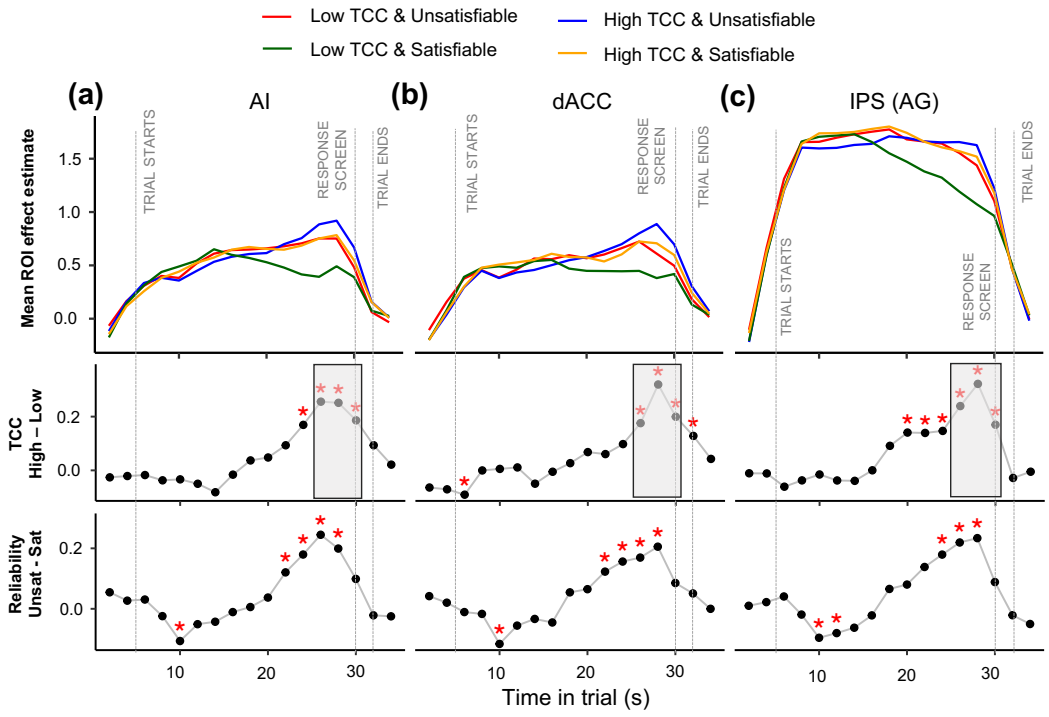
Interestingly, there seemed to be an interaction between satisfiability and TCC. Specifically, satisfiable instances with low TCC started showing a decrease in activity from early on in the trial in all three regions (Fig 5.2.6 green line). Conversely, unsatisfiable instances with high TCC showed a positive slope in both AI and dACC until late in the trial (orange line).

When contrasting the effect of TCC on each of the ROIs, we find that there is a significant positive effect of TCC from mid-way through the trial in the right IPS/AG (Fig 5.2.6 second row of panels). This differs from the results obtained from the whole brain analysis, which might be due to the increased power in ROI analyses. Similarly, when estimating the effect of satisfiability (Fig 5.2.6 third row of panels), the results marginally differ from those of the whole-brain analysis. Firstly, the ROI analysis reveals that there is an effect of satisfiability on all three regions late in the solving-stage. Secondly, the effect of satisfiability starts in the AI and dACC mid-way through the trial. Interestingly, the effect of TCC seems to precede

Table 5.2.3: **Accuracy clusters.** Significant cluster-wise FWE-corrected ($p < 0.05$) clusters (using an uncorrected threshold of $p < 0.001$) from the *Correct-Incorrect* contrast. Coordinates are in MNI space.

			Cluster statistics			Peak statistics			
Stage	Region	Side	Volume(mm^3)	β_{mean}	SEM	β_{peak}	x	y	z
S1	Occipital cortex	RH	569.3	0.21	0.003	0.31	10	-100	8
	AI	RH	2146.3	-0.22	0.002	-0.33	30	23	-8
	AI	LH	2048.0	-0.26	0.002	-0.40	-48	18	-12
	dACC	LH	1953.8	-0.24	0.002	-0.38	-2	22	40
	SFG	RH	1007.6	-0.18	0.003	-0.28	2	23	60
Response	MFG	LH	974.8	-0.19	0.002	-0.27	-27	50	16
	SFG	LH	684.0	-0.18	0.003	-0.26	-2	10	60
	Inferior frontal gyrus	RH	602.1	-0.22	0.003	-0.29	50	17	31
	ACC	LH	520.2	-0.18	0.002	-0.25	-3	31	26
	Paracingulate gyrus	LH	491.5	-0.24	0.004	-0.31	-5	9	50

Figure 5.2.6: **Temporal dynamics of regions of interest.** Mean effect estimate (β) of each ROI against time in trial. The effect at each time point represents the mean β_{FIR} over all of the voxels from each ROI: right AI **(a)**, dACC **(b)**, and right IPS cluster extending to the angular gyrus **(c)**. In the top row of figures, the β_{FIR}'s characterize the coefficients of the FIR regression with four conditions: satisfiability×TCC. The β_{FIR} parameters are aligned to the BOLD signal, which has a lag with respect to the task time. To correct for this, the gray vertical lines represent the task-events by assuming a 5 seconds BOLD signal lag. In the second row, the TCC contrast ($\beta_{high} - \beta_{low}$) is presented. The bottom row shows the satisfiability contrast ($\beta_{unsat} - \beta_{sat}$). Red asterisks represent significance at a 0.05 significance level. Significance levels in the gray shaded regions are suggestive only; they represent the time period and contrast from which the ROIs were selected.



that of satisfiability in the IPS, whereas in the dACC the effect of satisfiability seems to precede that of TCC.

Altogether, these results suggest that both satisfiability and TCC correlate with activity in all three regions, but that their effect might have different neural temporal signatures. Importantly, the sign of the effect was in line with our hypothesis; a higher signal in these regions was generally related to high TCC and unsatisfiability. The only exceptions happen briefly early in the trial, which might be related to evidence accumulation acting in these regions (see Discussion).

Psychophysiological interactions (PPI)

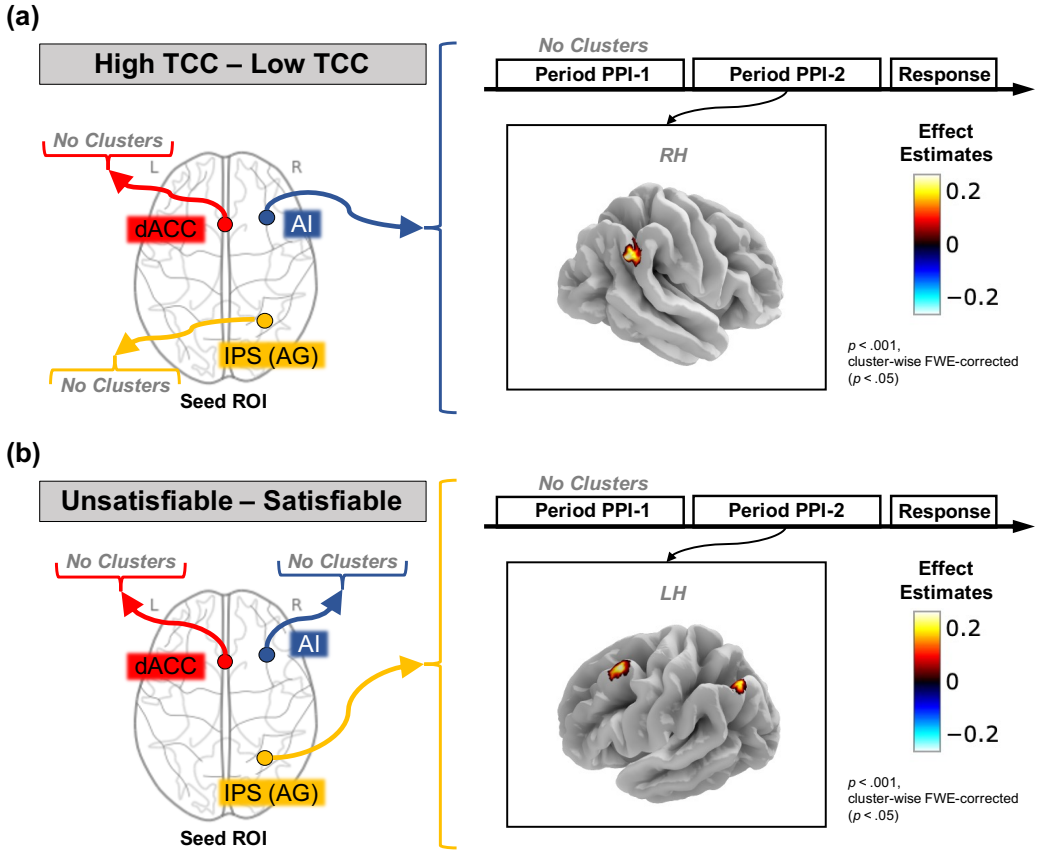
We were interested in studying the functional connectivity in the brain during problem-solving. Moreover, we wanted to assess the effect of the intrinsic properties of the problem on functional connectivity. To investigate these effects, we first conducted PPI analysis to gauge the functional synchronization between regions in the brain and each of the ROIs described in the previous section. Explicitly, we performed whole-brain PPI analyses employing the three relevant ROIs (dACC, rAG and rAI) as seed regions. For these regressions we modeled the task (items and solving stages together) with two boxcar functions of equal length (12.5s) (See Fig 5.2.7). This allowed us to study PPI task interactions separately for an early period (PPI-1: first 12.5 seconds of the trial) and a late period (PPI-2: last 12.5 seconds). We first explored the effect of each of these two task periods on the connectivity to each of the ROIs. We found a similar pattern of connectivity in all three ROIs, and both periods, when contrasting the PPI effect compared to baseline (see Fig C.1). Overall, these connectivity results show a reliable synchronization between each of the three seed ROIs with FPN and CON, during both periods. This suggest that the task has a similar effect on the BOLD synchronization pattern of all three regions.

Additionally, we investigated the differences between connectivity patterns for different types of instances. When comparing the connectivity between instances high TCC and low TCC we found one significant cluster with differential connectivity. This cluster, located along the rAG and the supramarginal gyrus, showed a change in connectivity to the rAI (seed region) between high and low TCC instances during the second PPI period (Fig 5.2.7a; Table 5.2.4). We then explored the differences in the PPI connectivity between unsatisfiable and satisfiable instances. We observed, a significant PPI effect of satisfiability between the right IPS/AG (seed) and the left MFG, as well as with the left AG, during the second PPI period (Fig 5.2.7b; Table 5.2.4). Overall, these results suggest that instance properties have an effect on the synchronicity between the ROIs and a limited collection of clusters. However, this effect is only significant during the later stage of the trial.

Table 5.2.4: **PPI clusters.** The effect of instances' properties on connectivity. Significant cluster-wise FWE-corrected ($p < 0.05$) clusters (using an uncorrected threshold of $p < 0.001$). Coordinates in MNI space.

			Cluster statistics			Peak statistics			
Contrast	Region	Side	Volume(mm^3)	β_{mean}	SEM	β_{peak}	x	y	z
TCC	AG/Supramarginal G.	RH	573.4	0.26	0.004	0.36	61	-50	36
Satisfiability	MFG	LH	499.7	0.300	0.007	0.475	-43	18	59
	AG	LH	483.3	0.266	0.006	0.432	-54	-65	47

Figure 5.2.7: **PPI results.** The effect of instances' properties on connectivity: **(a) TCC**, **(b) Satisfiability**. The left panel represents the seed region used for the analysis (dACC, rAG or rAI). The right panel shows the clusters that display a significant PPI connectivity effect for a particular seed region and period. *Significant cluster-wise FWE-corrected ($p < 0.05$) clusters (with an uncorrected threshold of $p < 0.001$) are presented. "No clusters": No significant clusters were found in this analysis.*



Granger causality analysis

PPI analysis provides a description of the functional connectivity (synchronization) between regions based on correlations between simultaneous activity across regions. As such, this analysis is insensitive to temporal directionality in the time series. In contrast, Granger causality (GC) is defined based on Vector Auto Regression (VAR) models, whereby a vector of ROI signals is driven by a finite number of lags of itself. This allows for gradual excitatory (or inhibitory) impact of one region onto another, that might suggest temporal directionality. This directional effect can be summarized by GC, which emerges when the presence of lags of one variable significantly improves the fit (maximum likelihood value) of another variable.

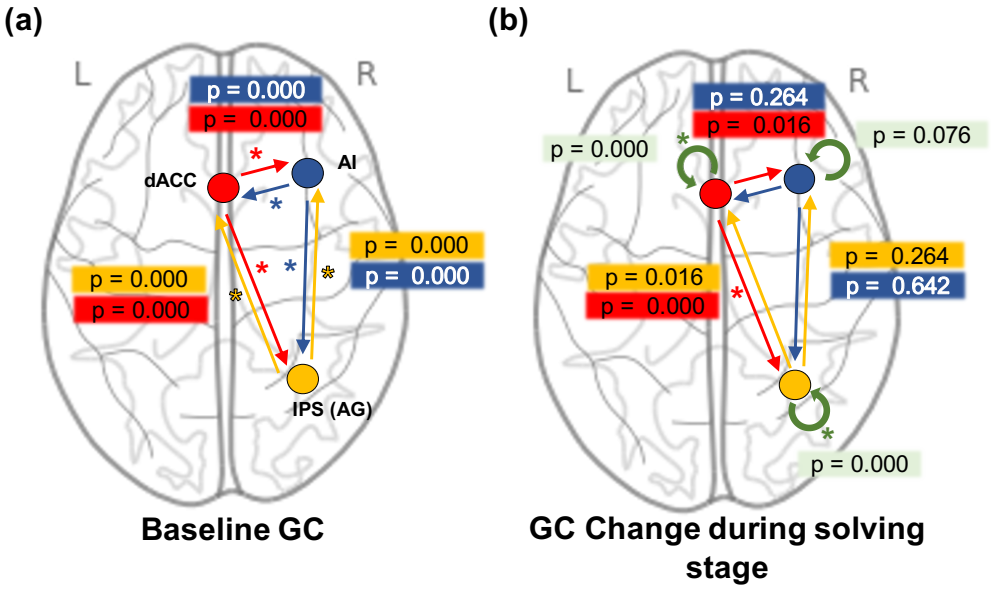
Critical for this study, we expected the underlying neural processes of complex problem-solving to be internally driven. Specifically, we expected the connectivity patterns to be linked to neural processes whose timing could vary stochastically across trials and participants (e.g., the burst of neural activity does not have to coincide with an experimental intervention such as initial display of items). In order

to explore these connectivity patterns we performed a GC analysis on the three ROIs. For this, we ran a VAR model on the ROI time series augmented with the series during the solving stage only, and determined incremental GC of one series on another during problem-solving. This allowed us to estimate effective connectivity GC tests at baseline as well as the GC changes from baseline during the solving stage.

We found a significant bidirectional connectivity between all the ROIs at baseline during the experiment (See Fig 5.2.8a). Additionally, during the solving stage we found that there was a significant increase in GC from dACC to rAG. In other words, the dACC (lagged time series) Granger-causes activation in rAG during the solving stage more intensely than elsewhere during the trial and inter-trial interval. Moreover, we found that during the solving stage there was an increased self-activation effect in the dACC and rAG; that is, the lagged time series of each of these two ROIs Granger-cause themselves (See Fig 5.2.8b).

We were also interested in the effect of instance properties on the effective connectivity between the 3 ROIs. To do this, we augmented our GC analysis to test for differences in connectivity between instances with varying intrinsic properties. Specifically, we compared connectivity across levels of TCC and satisfiability category. We did not find any significant changes in the effective connectivity between high and low TCC instances nor between unsatisfiable and satisfiable instances. A note of caution is due here since these results could be due to lack of statistical power.

Figure 5.2.8: **Granger causality results.** Effective connectivity estimated via Granger causality between each of three ROIs: dACC, rAI and rAG. **(a)** Represents the baseline connectivity between the regions. **(b)** Represents the changes in effective connectivity during the solving stage compared to baseline. Only three effects survive multiple comparisons correction: An increased connectivity from dACC to rAG and a higher self-modulatory effect on both dACC and AG. *P-values correspond to the GC test uncorrected for multiple comparisons. Asterisks represent significant GC effects FWE-corrected at significance threshold of 0.05.*



5.3 Discussion

The study of the neural underpinnings behind problem-solving has, to date, been centered on tractable problems. This line of research has led to the characterization of networks and processes associated with problem-solving. However, it remains an open question whether these results can be extended to complex problems. More fundamentally, it is not clear if and how the current theoretical framework can be used to study these problems. A critical complication in this gap is the difficulty of characterizing cognitive demand, which is particularly problematic in complex problems because of the plethora of strategies that might be employed (e.g., MacGregor and Chu 2011; Acuña and Parada 2010). In the present paper we propose a methodology to study the neural invariants of problem-solving in which, by focusing on the intrinsic properties of a problem, it is possible to characterize the cognitive demand of a task associated with computational hardness.

Employing this theoretical framework, we empirically studied the neural underpinning of complex problem-solving in the knapsack decision task using ultra-high field fMRI. Our findings shed light into the neural processes supporting problem-solving. Firstly, our findings not only extend but solidify the research on the neural correlates of cognitive demand by exploring the processes associated with one specific dimension of cognitive demand: computational hardness. Importantly, our results extend the study of the neural underpinnings of problem-solving by providing a framework for the study of intractable problems using a generic definition for cognitive demand. Secondly, the study of intrinsic properties of a problem and their connection to neural processes in problem-solving have significant implications for the understanding of how people solve these problems. In particular, using this approach is is possible characterize relevant neural markers of a task such as TCC and satisfiability. These markers might have significant implications on how people approach computational tasks, just like risk and variance have been shown to affect decisions in probabilistic tasks. Finally, the results presented here complement the investigation of cognitive control by providing a framework that can be employed to extend previous findings to tasks that involve intractable problems. Critically, cognitive control involves the dynamic allocation of cognitive resources that stem from an interaction between the cognitive requirements of a task and the resources available. The framework put forward here provides a theoretical foundation for the characterization of the former.

5.3.1 Neural correlates of cognitive demand

Extensive research has studied the neural correlates of cognitive demand. This program has characterized a MDS; a network of regions that respond to cognitive demand irregardless of the task at hand (Fedorenko, Duncan, and Kanwisher 2013; Assem et al. 2020; Duncan and Owen 2000; Crittenden, Mitchell, and Duncan 2016). This has been done using several tasks including perceptual target detection, memory retrieval, among many others. Notably, most of the tasks employed to date have been based on tractable problems. Moreover, many of the tasks employed modulate cognitive demand of the task by tuning the amount of processing needed on one specific dimension of cognitive processing. For instance in perceptual tasks signal to noise ratio is modulated (e.g., Aben et al. 2020; Dubis et al. 2016; Hanks and

Summerfield 2017; Ploran et al. 2011), alternatively, in memory retrieval tasks, the amount of information to be stored/retrieved is tuned (e.g., G. Gratton et al. 2018; Fedorenko, Duncan, and Kanwisher 2013).

The lack of a generic (problem-independent) definition of cognitive demand hinders the generalization of this approach to new problems. Importantly, the level of cognitive demand might be highly related to the strategies used. For instance, multiplication operations can be performed using different strategies such as the standard multiplication algorithm or alternatives such as the Japanese visual method and the Vedic method (Garain and Kumar 2018). These different strategies would generate different landscapes of cognitive demand in multiplication problems depending of the algorithm used. Critically, when leaping into tasks that are more complex, and especially those that involve problems that are intractable, the limitations of this approach become more apparent. The increase in complexity brings along an increase in strategies available to solve the problem and this makes the determination of a single metric of cognitive demand even more troublesome.

A proper quantitative (cardinal) study of the neural underpinnings of cognitive demand requires a proper generic definition of cognitive demand that can be quantified across problems and, ideally, across strategies. Such characterization would be grounded in the assumption that hardness is, at least partially, an intrinsic characteristic of the problem at hand. Here we take this approach and present a framework that builds on computational complexity theory and is able to categorize problems in a generic way according to their intrinsic computational hardness. We show that computational hardness, as so defined, can be used to study to the neural correlates of cognitive demand.

We empirically studied and identified the neural correlates of computational hardness in the knapsack decision task. Specifically, we found that the neural correlates of TCC overlapped with those associated to the MDS. In particular, the positively correlated clusters (higher activation in high TCC instances) in the FPN and CON resembled those of the MDS. Notably, we found clusters in the AI, the dACC, the precentral gyrus and the IPS, which are ascribed to the MDS (Fedorenko, Duncan, and Kanwisher 2013). Moreover, we found clusters in the occipital lobe. Although occipital regions are not generally assigned to the MDS, they do show a differential activation when modulating cognitive demand in several tasks (Fedorenko, Duncan, and Kanwisher 2013). Importantly, our results display a dynamic process in which the neural correlates of TCC vary throughout the different stages of the task. These clusters varied in their location and their correlation with TCC. Specifically, the positively correlated clusters were found only from halfway through the solving stage. This suggests that the effects of computational hardness on calculation and control allocation are only realized late in the trial. Moreover, activity in these clusters did not show a sustained significant correlation with TCC; they changed throughout the late stages of the trial. Of note, the effect size of the correlation with TCC and the three ROIs considered changed between period three and four of the solving stage. This suggests that the MDS can be construed as a heterogeneous set of regions that play a dynamic and varying role at different stages in problem-solving.

5.3.2 Task-related neural markers

Here we extend the study of the neural underpinnings of problem-solving to a canonical intractable (NP-hard) problem: the knapsack problem. Importantly, we do this by exploring the link between generic properties of the problem and their link to neural processes. In the previous section we hinted at a direct effect of cognitive demand on the neural computations needed to solve the problem. A related effect of these properties on neural processes is through the encoding of relevant task markers that could be employed during problem-solving (Yoo, Hayden, and Pearson 2021; Koechlin 2016).

These neural markers include markers of performance such as expected error (Neta, Steven M Nelson, and Petersen 2017; Bossaerts 2018), variance in this expectation (uncertainty) (Neta, Steven M Nelson, and Petersen 2017; Neta, Schlaggar, and Petersen 2014; Bossaerts 2018), as well as markers that encode the evidence towards a particular response (Ploran et al. 2011; C. Gratton et al. 2017) or even the merit of alternative strategies (Duverne and Koechlin 2017; Donoso, Collins, and Koechlin 2014). Critically, we hypothesized to see markers of performance, related to TCC, from early on in the trial. Additionally, we conjectured we would see markers of reliability in regions shown to encode uncertainty. Finally, we expected to find neural correlates of accuracy late in the trial, which would be associated to expected performance.

We expected to see neural correlates of TCC from early on in the trial. TCC is a feasible metric that can be related to markers of performance and efficacy of effort from early on in the trial because it stems from constrainedness, which can be potentially estimated from early on during the solving stage. Indeed, estimation of constrainedness by the agent is feasible without the need to know the solution to the problem because constrainedness (and thus TCC) can be potentially estimated by performing a sum and division operations (e.g., $\alpha_p = p/\sum_{i=1}^{N} v_i$). Moreover, this metric has been demonstrated to be correlated to human performance. We specifically, expected to see markers of TCC in the CON (Shenhav, Botvinick, and J. D. Cohen 2013; Bossaerts 2018; Neta, Schlaggar, and Petersen 2014). However, contrary to our expectations, we only found significant clusters in the CON starting from the third period of the solving stage. These might reflect markers of expected performance, but other explanations cannot be excluded. For instance, this effect might reflect differences in time-on-task between TCC conditions (Grinband et al. 2011). This explanation, however, would still allow these activation patterns to represent differences in neural markers such as reliability and expected performance. This follows from the fact that time-on-task is an endogenous variable of the system. That is, the agent decides when to stop, and as such, this decision would follow from a subjective belief on how well they can expect to perform given the current candidate solution. Therefore, differences in time-on-task between high and low TCC instances would probably entail differences in subjective beliefs of both expected performance and reliability.

Besides the reported clusters that correlated positively with computational hardness, we found a set of clusters that correlated negatively with TCC. These clusters are concentrated in the second period of the solving stage, but are also found on the third and fourth periods of the solving stage. These results might be explained by the encoding of evidence accumulation signals (Ploran et al. 2011). Arguably, evidence toward a solution can be accumulated faster in low TCC compared to high

TCC instances. This would imply that regions that encode evidence accumulation would show a higher activation on low TCC instances early in the trial, in accordance to the pattern found on the second period of the solving stage.

In addition to studying markers linked to TCC, we explored the correlates of satisfiability during problem-solving. We expected to see activation related to satisfiability in regions previously associated with uncertainty encoding; specifically in the CON. In line with our hypothesis we found a significant positive relation between unsatisfiability and activity in the CON that started halfway through the trial. Contrary to our expectations we found several regions that displayed an increase in activity during satisfiable instances from early on in the solving stage. This result is perplexing because knowing the satisfiability of the problem equates to having solved the problem, which would not be expected early on in the trial. A possible explanation for this is that the clusters found encode evidence accumulation (Ploran et al. 2011; C. Gratton et al. 2017) and that accumulating evidence towards the solution in satisfiable instances occurs at a different rate than in unsatisfiable instances. Relatedly, these activation patterns might reflect the use of different strategies. However, this account would still require participants to be implementing different strategies, based on satisfiability, as early as during the first few seconds of the solving stage.

Additionally, we studied error related signals by studying the effect of accuracy on the neural activation throughout the task. It has been shown that the FPN and CON encode task signals related to error detection and error expectation (Neta, Steven M Nelson, and Petersen 2017; Neta, Schlaggar, and Petersen 2014; Dosenbach, Visscher, et al. 2006). Although participants did not receive any feedback during the task, we expected to see error related markers during late stages of the trial. Although these signals would not represent the integration of novel exogenous information, we hypothesized that participants would represent a subjective judgment on the perceived correctness of their solution (Duverne and Koechlin 2017). The results of this analysis confirm our hypothesis by outlining a set of regions in both FPN and CON that encode errors during the response stage. In contrast, this analysis did not result in any other clusters during the solving stage that correlated negatively with accuracy. The lack of negative correlation during the late periods of the solving stage could be due to variability in the signal during the solving stage. Indeed, during this stage participants might be updating their accuracy expectation as well as their candidate response. Since our accuracy contrasts is based on the answer provided during the response stage, it stands to reason that our analysis does not capture accuracy markers during the solving stage because we do not have a measure of accuracy during this period. It is worth noting that we found one significant cluster during the solving stage (in period one) that correlated positively with accuracy in the occipital cortex. This could reflect attentional differences, early on in the trial, which affect performance in the trial.

Taken together, the framework put forward here provides a way to study neural markers associated to subjective beliefs during complex-problem-solving. Importantly, this is done using a generic framework that allows for the characterization of markers across problems without the need to consider the strategies implemented. Overall, we found evidence that suggests the existence of neural markers related to computational hardness, reliability and performance during complex problem-solving. Importantly, we find markers of computational hardness and reliability by

analyzing the intrinsic features of the task. In line with our conjecture, we found that the neural markers of the reliability overlap with regions that encode probabilistic uncertainty. This suggests that reliability and uncertainty might constitute analogous constructs that are encoded similarly across tasks and that could serve a generic role in decision-making.

5.3.3 Computational hardness and cognitive control in problem-solving

The generic framework put forward in this manuscript is particularly valuable in the study of allocation of control during problem-solving given the robustness of this process. The human brain has limited cognitive resources, yet is capable to reuse and reallocate resources in order to successfully perform plethora of cognitive tasks. Critically, it is able to perform tasks that involve solving problems that are deemed hard or complex. A true understanding of the human brain and its neural processes would require a generic model capable of generalizing across the specifics of a task. In this line, several proposals have been put forward in which the allocation of limited cognitive resources is modeled as a mechanism in which a generic currency of cognitive demand is estimated and effort (or control) is consequently allocated based on this characterization (e.g., Shenhav, Botvinick, and J. D. Cohen 2013; Verguts, Vassena, and Silvetti 2015; Westbrook and Braver 2015). Such currency would probably need to be characterized from generic features of the task such that the agent is able to estimate cognitive demand across tasks. Here, we define a potential component of this currency: computational demand. This dimension of cognitive demand captures the amount of computations needed to solve a problem based on insights from computational complexity theory. Importantly, this approach is readily generalizable to several other complex problems (Cheeseman, Kanefsky, and W. M. Taylor 1991; Percus, Istrate, and Moore 2006; Ian P. Gent et al. 1996; Yadav et al. 2020; Franco, Doroc, et al. 2021) without the need to assume a particular procedural strategy used to solve a problem. This can inform the study of cognitive resource allocation in order to generalize patterns across problems and idiosyncratic strategies.

In order to explore the dynamics related to control during complex problem-solving we analyzed the functional interaction during problem-solving of three ROIs. Two associated with cognitive control (i.e., CON) and one region associated with processes that were deemed highly relevant for the task at hand (i.e., IPS). We studied synchronization of signals (employing a PPI analysis) and explored their effective connectivity (using GC analysis).

PPI results showed a generalized change in signal synchronization during the solving stage compared to baseline. Moreover, when exploring the link between instance properties and synchronicity between regions we found a few clusters whose connectivity was modulated by either satisfiability or TCC. These effects were only present late in the trial. Specifically, we found that TCC modulated the synchronicity between the rAI and the rIPS. Additionally, satisfiability modulated the functional connectivity between the right IPS and two clusters in the left hemisphere, one in the AG and one in the MFG. Overall, these results suggest a differential recruitment of regions during the task that is, partially, modulated by task properties late in the trial. Interestingly, the significant clusters identified in this analysis

have been implicated in the performance of mathematical calculations (Arsalidou and M. J. Taylor 2011; Grabner et al. 2009), suggesting that they could support moment-to-moment implementation of strategies. Further work would be needed in order to asses whether the relation, found here, between instance properties and functional synchronization is associated to the implementation of different strategies.

Additionally, we explored the effective connectivity between the three ROIs described. We found that there was higher effective connectivity from dACC to IPS during the solving stage of the task. These results extend those previously found in perceptual tasks (Aben et al. 2020), in which regions relevant for the task at hand showed a higher functional connectivity to the dACC during the task. Overall, this result further supports previous research that assign the dACC a central role in the allocation of control (Shenhav, Botvinick, and J. D. Cohen 2013; Dosenbach, Visscher, et al. 2006; Silvetti et al. 2018; Vassena, Holroyd, and Alexander 2017; Holroyd and Yeung 2012; Alexander and Brown 2011; Sestieri et al. 2014; Aben et al. 2020; Crottaz-Herbette and Menon 2006).

Interestingly, we did not find a significant increase in the effective functional connectivity between rAI and rIPS during the solving stage. This finding matches previous research that support a dissociation between dACC and AI (Han, Eaton, and Marois 2019; Steven M. Nelson et al. 2010; Vinod Menon and Uddin 2010; Wu et al. 2019). However, these results seem to be contrary to those found by Sestieri et al. 2014, who found increased functional connectivity between AI and task-relevant regions in perceptual and episodic memory tasks. Several possible explanations could be put forward to account for this discrepancy. For instance, the nature of the functional connectivity between rAI and task-relevant regions might be task-specific. Specifically, it has been suggested that AI is predominantly involved in processing of internal visceral and motivational information involved in autonomic behavior (Steven M. Nelson et al. 2010). This type of processing might be more relevant in perceptual and episodic memory tasks compared to the knapsack task, in which mathematical calculations might be more pertinent. Alternatively, other possible explanations for the lack of significant effective connectivity between rAI and rIPS include the lack of statistical power in this analysis, as well as discrepancies in the ROI definition.

Another significant aspect considered was the link between effective connectivity and the intrinsic properties of the instance at hand. Our results suggest that the effective connectivity pattern was impervious to the level of computational demand and satisfiability. Of particular relevance, we found that the effective connectivity between dACC and IPS was not modulated by TCC. This suggests that the effect of TCC on control, if any, occurs by generating differential levels of activity within the regions of interest and not via modulation of the functional connectivity between these regions. Of course, this failure to reject the null hypothesis could be due to a lack of power or the exclusion of relevant ROIs from the analysis. We leave it to future research to explore how whole brain connectivity patterns are affected by computational demand.

Overall, the framework presented in this paper allows for the study of control in complex problem-solving. Here we applied this framework to a complex problem and explored how it could be used to elucidate our understanding of cognitive control. Notably, we showed that the CON and the IPS are synchronized with a several regions during task performance. Moreover, we found that dACC had a higher

directional connectivity with a task-relevant region (IPS) during problem-solving.

Our approach differs in many ways with the more commonly used tasks in the study of cognitive control. Notably, commonly used tasks in cognitive control are highly process specific. That is, problem-solving in these tasks involve precise sub-processes, thus sacrificing environmental validity for specificity. Consider, for instance, four prominent categories of tasks used to study and manipulate the level of cognitive control: task switching tasks, conflict tasks, inhibition tasks and working memory tasks (see G. Gratton et al. 2018 for a review). In each of these tasks, the neural correlates of cognitive demand are associated to a particular sub-process such as memory retrieval (e.g., N-back task), withholding of a prepotent tendency (e.g., Stroop task, go/no go task) or switching between task-sets. The approach introduced in this manuscript presents a way forward to include into the scope of the analysis the interplay of these sub-processes by using more environmentally valid tasks.

It is worth highlighting that we are not arguing for the proposed framework to replace other methodological approaches in the study of cognitive control. Instead, we assert that both approaches complement each other. Critically, complex tasks involve the interplay of several computational processing units such as working memory, logical operations, processing of numerical magnitudes among many others. Our approach, as it stands, is not able to differentiate among these sub-processes. A proper understanding of complex problem-solving requires both the study of these sub-processes independently, like in more classical approaches (G. Gratton et al. 2018), as well in tandem, like done in this paper.

5.3.4 Directions for future research

In the present paper we propose a methodology through which, by focusing on the intrinsic characteristics of a complex computational task, it is possible to characterize neural invariants of complex problem-solving. This resembles prominent approaches in the study of problem-solving in other types of tasks. Notably, in perceptual tasks the neural underpinning have been explored by studying the difference in neural processes with respect to intrinsic features of the task. For instance, many tasks explore the neural processes behind perceptual identification of a target. Neural processes are identified by contrasting the neural activity between conditions with varying levels of perceptual signal strength (e.g., Dubis et al. 2016; Hanks and Summerfield 2017; Ploran et al. 2011). Relatedly, in probabilistic tasks, invariants are captured by studying the neural processes linked to intrinsic characteristics such as mean, variance and related stochastic metrics (e.g., Preuschoff, Bossaerts, and Quartz 2006; D'Acremont and Bossaerts 2016; D'Acremont and Bossaerts 2008; d'Acremont, Schultz, and Bossaerts 2013; Christopoulos et al. 2009; O'Neill and Schultz 2013).

By applying this framework to the study of complex tasks, we identified neural invariants of problem-solving in a generic way. That is, like in perceptual and probabilistic tasks, the relevant features (e.g., computational hardness) can be studied across tasks. Indeed, TCC has previously been shown to affect human behavior in tasks involving other NP-complete problems (Franco, Doroc, et al. 2021). This framework represents a new approach in the study of problem-solving, by characterizing a scale of cognitive demand that can be compared across problems. This allows

for the study of the generic (across problems) neural substrates of problem-solving associated to computational demand. Future work should extend the results, beyond the knapsack problem, to other complex problems in order to characterize a truly task-independent core of controllers and processors supporting complex computations in the brain.

In this paper, we studied how neural processes were affected by modulating one significant dimension of cognitive demand: computational hardness. We specifically explored the computational hardness associated to the constrainedness of an instance. This dimension is a fundamental source of cognitive demand when solving complex problems. Not only has it been shown to be associated with the computational requirements of solving problems by several algorithms (Cheeseman, Kanefsky, and W. M. Taylor 1991; Achlioptas, Naor, and Peres 2005; Selman and Kirkpatrick 1996; Yadav et al. 2020; Ian P Gent and Walsh 1996; Ian P. Gent et al. 1996), but it has been shown to affect human decision quality in the knapsack decision problem and other NP-complete problems (Franco, Yadav, et al. 2020; Franco, Doroc, et al. 2021). Notwithstanding its relevance, this source of cognitive demand is not unique. Other sources of cognitive demand have been identified to affect human problem-solving. Notably, it has been shown that the quality of the solution decreases as the size of the problem increases (e.g., Carruthers, Masson, and Stege 2012; MacGregor and Chu 2011; Dry et al. 2006; van Opheusden and Ma 2019; Stazyk, Ashcraft, and Hamann 1982; De Visscher and Noël 2014). Additionally, it has been demonstrated that problem-specific and strategy-specific features of an instance affect the quality of decisions (e.g., Murawski and Bossaerts 2016; MacGregor and Chu 2011; Basso, Bisiacchi, et al. 2001; Basso and Saracini 2020). Further work in this area is needed to understand the interaction between different sources of cognitive demand on human problem-solving.

We have investigated complex problem-solving by employing a theoretical framework that examines the computational hardness of random ensembles of instances of NP-hard problems. TCC quantifies average complexity over a set of random instances conditional on a fixed property (α). This is particularly relevant for the study of cognition because it is a metric that can be estimated without the need to solve the problem. In contrast, many alternative metrics of difficulty require knowledge of the solution, which makes them computationally expensive. This includes IC, the number of solution witnesses, as well as other strategy-specific metrics such as Sahni-K (Murawski and Bossaerts 2016; Sahni and Sartaj 1975). Despite the advantages of TCC with regards to its computational feasibility, it is contingent on a random generation process. Here we sampled instances, specifically, from a uniform distribution. Future work could aim at characterizing real-life distributions of instances and whether the findings presented here are robust to these distributions.

In this study we studied the neural correlates of the knapsack *decision* task, a task associated to an NP-complete problem. In general, the theoretical framework presented here is applicable to decision problems, whereas many tasks encountered in real life might involve optimization problems. It has been shown that this framework can be extended to optimization problems in which finding the exact optimum is required (Franco, Yadav, et al. 2020). However, finding the exact solution might not always be essential in the real-world. In many cases, finding an approximate or 'good enough' solution might suffice. The framework put forward in this paper provides a direct way for studying the computational demands of the latter. Indeed,

several accounts of decision-making, model humans as satisficing agents. That is, an agent whose objective is not to find the optimum (e.g., find items that maximize value in the knapsack), but to reach a target with 'good enough' value (e.g., find a selection of items that reaches a target profit) (Herbert A Simon 1956; Bossaerts 2018). Future research could investigate whether the results found in this study can be extended to other types of approximation schemes.

Since many of the decision-making tasks faced on a daily basis involve solving complex problems, it follows that the understanding of real life human decision-making requires a comprehensive understanding of how people solve these problems. This is particularly relevant for the investigation of real-life cognitive deficiencies and the closing of the gap between deficiencies reported in lab settings (tractable problems) and those present in real-life (intractable) situations (see Bielak, Hatt, and Diehl 2017 for a review of this gap in the context of aging). Our results showing a lack of significant correlations between the measured cognitive abilities and performance in the knapsack tasks alludes to this gap. Further research could, for instance, utilize this framework to investigate how compensatory mechanisms identified during cognitive decline (e.g., López-Góngora et al. 2015; Kaufmann et al. 2009; Sala-Llonch, Bartrés-Faz, and Junqué 2015) support complex problem-solving as well as when and how they cease to work. The increased ecological validity of complex problems could provide insights into when cognitive decline is expected to hinder real-life decisions.

Humans are constantly solving problems. From perceptual tasks, such as motion detection and face recognition, to more complex tasks such as choosing an investment portfolio. Understanding the neural processes involved in this core function of the brain is one of crucial importance for the understanding of human decision-making. Here we presented a framework that allows for the study of human complex problem-solving. We applied this framework to the study of the neural underpinnings of problem-solving and identified a dynamic set of regions that respond to cognitive demand when performing complex tasks. Overall, the findings from this manuscript provide support for the premise that computational hardness, as described in this paper, is a fruitful characterization of cognitive demand of complex problems for neuroscience. This calls for a closer collaboration between cognitive neuroscientists and computer scientists for the successful advancement of the field of problem-solving in both human and electronic computers.

5.4 Materials and methods

5.4.1 Ethics statement

The experimental protocol was approved by the University of Melbourne Human Research Ethics Committee (Ethics ID 1749616.3). Written informed consent was obtained from all participants prior to commencement of the experimental sessions. Experiments were performed in accordance with all relevant guidelines and regulations.

5.4.2 Participants

Twenty right-handed volunteers from Melbourne University and the surrounding community took part in the study (14 female, 5 male, 1 other; age range = 18-35 years, mean age = 26.6 years). Inclusion was based on age (minimum = 18 years, maximum = 40 years) and on right-handedness. Each participant performed the knapsack decision task in the scanner and performed outside the scanner the knapsack optimization task, a mental arithmetic task and a set of basic cognitive function tasks.

5.4.3 Knapsack decision task

In this task, participants were asked to solve a number of instances of the (0-1) knapsack decision problem (Fig 5.2.1). In each trial, they were shown a set of items with different values and weights as well as a capacity constraint and a target profit. Participants had to decide whether there exists a subset of those items for which (1) the sum of weights is lower or equal to the capacity constraint and (2) the sum of values yields at least the target profit.

Each trial had four stages. In the first stage (items stage; 3 seconds), only the items were presented. Item values, in dollars, were displayed using dollar bills and weights, in grams, were shown inside a black weight symbol. The larger the value of an item, the larger the dollar bill was in size. Similarly, the larger the weight of an item, the larger its weight symbol was in size. At the center of the screen, a green circle indicated the time remaining in this stage. In the second stage (solving stage; 22 seconds), target profit and capacity constraint were added to the screen inside the green timer circle. In the third stage (response stage; 2 seconds), participants saw a 'YES' and a 'NO' button on the screen, in addition to the timer circle, and made a response using the keyboard (Fig 5.2.1). Finally, a jittered inter-trial rest period of 8, 10 or 12 seconds was shown before the start of the next trial.

Participants completed 56 trials (7 blocks of 8 trials), each showing a different instance of the knapsack decision problem. The order of instances was randomized across participants. The side of the 'YES' and 'NO' buttons was also randomized.

Instance properties and complexity

In the present paper, we examine the neural correlates of cognitive demand in a complex problem-solving task employing a metric of generic hardness that stems from computational complexity theory. A notable approach in this theory studies computational hardness by analyzing the difficulty of randomly generated instances of *decision problems*. That is, those problems whose answer is either 'yes' or 'no'. The study of random ensembles of instances has characterized a source of considerable variation in computational hardness for instances with the same input length (Cheeseman, Kanefsky, and W. M. Taylor 1991; Percus, Istrate, and Moore 2006; Ian P. Gent et al. 1996; Yadav et al. 2020). This variability in hardness has been related to various structural properties of instances. In particular, it has been shown that for several NP-complete problems there exist a set of parameters $\bar{\alpha}$ that captures the constrainedness of an instance; that is, the likelihood that the problem is *satisfiable* (i.e., the solution to the problem is 'yes') (Cheeseman, Kanefsky, and W. M. Taylor 1991; Ian P. Gent et al. 1996; Ian P Gent and Walsh 1996; Yadav et al.

2020). This line of research has found that there is a threshold α_s such that random instances around these threshold are harder than instances further away from it (Cheeseman, Kanefsky, and W. M. Taylor 1991; Krzakala et al. 2006; Ian P Gent and Walsh 1996; Achlioptas, Naor, and Peres 2005; Selman and Kirkpatrick 1996; Krzakala et al. 2006). We call this source of complexity, typical-case complexity (TCC).

In the knapsack problem, TCC is explicitly connected to a set of parameters $\bar{\alpha} = (\alpha_c, \alpha_p)$ that capture the constrainedness of the problem: $\alpha_p = \frac{p}{\sum_{i=1}^{N} v_i}$ and $\alpha_c = \frac{c}{\sum_{i=1}^{N} w_i}$ (Yadav et al. 2020; Franco, Yadav, et al. 2020). These parameters determine the likelihood that a random instance is *satisfiable*. Specifically, they characterize where typical instances are generally satisfiable (under-constrained region), where they are unsatisfiable (over-constrained region) and where the probability of satisfiability is close to 50% (satisfiability threshold α_s). TCC is defined based on the distance to the satisfiability threshold. Specifically, instances with values of α_p near the satisfiability threshold have a high typical-case complexity (*high TCC*) whereas instances further away from it—that is, in the under-constrained and over-constrained regions— have low typical-case complexity (*low TCC*). Importantly for our study, TCC has been shown to affect human behavior in the knapsack task (Franco, Yadav, et al. 2020).

In our analyses, we also explored the effect of two other intrinsic properties of instances of the knapsack decision problem: satisfiability and instance complexity (IC). Satisfiability refers to the solution of an instance; explicitly, if the solution to an instance is 'yes' it is *satisfiable*, otherwise it is *unsatisfiable*. This property, affects how a problem can be solved. On the one hand, to prove that an instance is satisfiable, a single solution witness needs to be found that satisfies both capacity and value constraints. On the other hand, to demonstrate that an instance is unsatisfiable all possible item combinations must be shown to be unable to satisfy the constraints. This can be done, for instance, via exhaustive search or logical pruning of alternatives.

The other intrinsic property, IC, is defined as the distance between the level of the profit constraint (target profit) and the maximum value attainable in the corresponding instance of the optimization variant of the 0-1 knapsack problem. Specifically,

$$IC = \left| \frac{p - p^{\star}}{\sum v_i} \right| = \left| \alpha_p - \alpha_p^{\star} \right|, \tag{5.1}$$

where p is the target profit of the decision instance and $p^{\star}$ is the maximum value achievable in the corresponding optimization instance, that is, the maximum value that can be packed into the knapsack given the same set of items I and the same capacity constraint c. α_p and $\alpha_p^{\star}$ denote the normalized values of target profit and optimum value, respectively.

Note that both TCC and IC capture the hardness related to constrainedness. However, they do so at two different levels of analysis. TCC, on the one hand, is a metric that captures the expected (average) difficulty of an ensemble of instances. On the other hand, IC is a metric of complexity of a single instance. Critically, while TCC can be estimated from the features of the problem alone, in order to estimate IC the optimization variant of the problem needs to be solved first. In that regard, TCC is a *feature-space ex-ante metric* (can be estimated before solving the

problem) whereas IC is an *ex-post solution-space metric* that can only be estimated after the problem has been solved. It is worth noting, however, that while TCC is less computationally intensive, it depends on the random generation process of instances.

5.4.4 Instance sampling

Sampled instances generate a 2×2 balanced factorial design of TCC (high vs. low) and satisfiability (satisfiable vs. unsatisfiable) with 18 instances in each condition.

Instances selected were a sub-sample of the instances used by a previous behavioral study (Franco, Yadav, et al. 2020). Instances in their study were selected such that α_c was fixed ($\alpha_c \in [0.40, 0.45]$) and the instance constrainedness varied according to α_p. 18 instances were selected from the under-constrained region ($\alpha_p \in [0.35, 0.4]$; *low TCC*) and 18 from the over-constrained region ($\alpha_p \in [0.85, 0.9]$; *low TCC*). Additionally, 18 satisfiable instances and 18 unsatisfiable instances were sampled near the satisfiability threshold ($\alpha_p \in [0.6, 0.65]$; *high TCC*). Half of the instances with high TCC were forced to have high/low computational requirements (top/bottom 50%), according to an algorithm-specific ex-post complexity measure of a widely-used algorithm (Gecode ;Gecode Team 2006). All instances in the experiment had $N = 6$ items and w_i, v_i, c and p were integers.

In the current study we randomly selected 56 of the 72 instances sampled in Franco, Yadav, et al. 2020. Sub-sampling without replacement was done ensuring that the same number of instances were selected across TCC and satisfiability conditions. Moreover, instances with high TCC were balanced to require high/low computational requirements according to the same algorithm-specific complexity measure employed in their study (i.e., Gecode propagations).

5.4.5 Complementary tasks

Participants were presented a set of complementary tasks outside of the scanner. They were asked to solve a number of instances of the (0-1) knapsack optimization problem. Similar to the knapsack decision task, participants were shown a set of items with different weights and values as well as a capacity constraint. However, unlike the decision variant, no target profit was presented. Participants had to find the subset of items that *maximized* total value subject to the capacity constraint (see Appendix A.1).

Additionally, we tested participants' performance on five aspects of cognitive function that we considered relevant for the knapsack tasks, namely, working memory, episodic memory, strategy use, processing and psychomotor speed, as well as mental arithmetic. To do so, we first administered a set of tasks from the Cambridge Neuropsychological Test Automated Battery (CANTAB; see Appendix A.2). Specifically, we asked participants to perform the Reaction Time (RTI), Paired Associates Learning (PAL), Spatial Working Memory (SWM) and Spatial Span (SSP). In addition, to test arithmetic abilities, participants were presented with a set of mental arithmetic problems (see Appendix A.2).

5.4.6 Procedure

Participants were asked to fill in an MRI screening form before attending the experiment. Once at the experiment, participants were presented with a plain language statement and a consent form. After reading these and providing written informed consent, participants were instructed in the tasks and completed a practice session of the knapsack decision task. Participants then underwent an MRI safety check and debriefing.

Before being scanned, participants solved the CANTAB RTI task outside of the scanner. This was followed by the scan session in which they performed the knapsack decision task. Afterwards, outside of the scanner, they completed the CANTAB RTI task again, followed by the knapsack optimization task. Subsequently, they completed the remaining CANTAB tasks in the following order: PAL, SWM and SSP. Finally, they performed the mental arithmetic task and completed a set of demographic and debriefing questionnaires. Altogether, the experimental session lasted around three hours.

Participants received a show-up fee of A$10, as well as monetary compensation based on performance. They earned A$1.2 for each correct answer in the knapsack decision task and for each correct answer in the knapsack optimization task.

5.4.7 Behavioral statistical analyses

The R programming language was used to analyze the behavioral data. All of the linear mixed models (LMM), generalized logistic mixed models (GLMM) and censored linear mixed models (CLMM) included random effects on the intercept for participants (unless otherwise stated). Different models were selected according to the data structure. GLMM were used for models with binary dependent variables, LMM were used for continuous dependent variables and CLMM were used for censored continuous dependent variables (e.g., time-on-task).

All of the models were fitted using a Bayesian framework implemented using the probabilistic programming language Stan via the R package 'brms' (Bürkner 2017). Default priors were used. All population-level effects of interest had uninformative priors; i.e., an improper flat prior over the reals. Intercepts had a student-t prior with 3 degrees of freedom and a scale parameter that depended on the standard deviation of the dependent variable after applying the link function. The t-student distribution was centered around the mean of the dependent variable. Sigma values, in the case of Gaussian-link models, had a half student-t prior (restricted to positive values) with 3 degrees of freedom and a scale parameter that depended on the standard deviation of the dependent variable after applying the link function. Standard deviations of the participant-level intercept had a half student-t prior that was scaled in the same way as the sigma priors.

Each of the models presented was estimated using four Markov chains. The number of iterations per chain was by default set to 2000. This parameter was adjusted to 4000 on some models to ensure convergence, which was verified using the convergence diagnostic $\hat{R}$. All models presented reach an $\hat{R} \approx 1$.

Statistical tests were performed based on the 95% credible interval estimated using the highest density interval (HDI) of the posterior distributions calculated via the R package 'parameters' (Lüdecke, Ben-Shachar, and Makowski 2020). For each

statistical test we report both the median ($\beta_{0.5}$) of the posterior distribution and its corresponding credible interval ($HDI_{0.95}$).

No participant nor trial was excluded from the data analysis of the knapsack decision task.

5.4.8 MRI data acquisition

We collected the fMRI images using a 7 Tesla Siemens MAGNETOM scanner located at the Melbourne Brain Centre (Parkville, Victoria) with a 32-channel radio frequency coil.

The BOLD signal was measured using a multiband echo-planar imaging sequence (TR = 800 ms, TE = 22.2 ms, FA = 45°). We acquired 84 interleaved slices (thickness = 1.6 mm, gap = 0 mm, FOV = 208 mm, matrix = 130x130, multi-band factor = 6, voxel size=1.6×1.6×1.6mm^3) per volume. 380 volumes were acquired on each run while recording cardiac and respiratory traces.

After five functional runs (one resting state run followed by four task runs), a high resolution (0.7 mm isotropic) anatomical image was acquired using an MP2RAGE pulse sequence (TR=5000 ms, TE=3.07 ms, TI1 = 700ms, FA1 = 4°, TI2 = 2700ms, FA1 = 5°, matrix=330×330, voxel size=0.73×0.73×0.73mm^3, FOV=240 mm, 224 slices, slice thickness = 0.73). Afterwards, another three functional runs were performed, followed by a diffusion weighted imaging (DWI) multi-band sequence (TR=7000 ms, TE=72.4 ms, FA =90°, FoV = 210 mm, matrix = 170x170, slice thickness =1.24, voxel size = 1.24m^3, 128 slices, multi-band factor =2).

5.4.9 Imaging statistical analyses

Preprocessing

Initial preprocessing of the data was performed using AFNI (Cox 1996) and the Advanced Normalization Tools (ANTs) software. For each subject, pulse and cardiac noise was regressed out from the functional scans. These were then slice-time corrected and the volumes were motion-corrected by registering them to the first volume of the first functional run. The mean image of the first run was co-registered to the anatomical scan (down-sampled) and this transformation was applied to all of the functional volumes. Afterwards, each participant's anatomical scan was used for calculation of transformation parameters to normalize the functional images into the Montreal Neurological Institute (MNI) space (see Appendix B for more details).

Whole-brain analysis (boxcar)

Whole-brain analyses were performed by fitting generalized linear models (GLM) using AFNI (Cox 1996). Before the regressions were implemented, we spatially smoothed the functional volumes with a 4.8mm FWHM Gaussian kernel. Additionally, volumes with motion or signal outliers were censored from each of the regressions.

We performed GLM regressions to explore three contrasts of interest. Specifically, we tested the neural correlates of TCC (high TCC vs. low TCC), satisfiability (unsatisfiable vs. satisfiable) and accuracy (correct vs. incorrect). In each of the

regressions the solving phase (22s) was modeled using four boxcar functions of equal duration (5.5s):

$$y = \beta_0 + \sum_{i=1}^{4} \left[\beta_i^{L0} L_0 \times box_{Si} + \beta_i^{L1} L_1 \times box_{Si}\right] + \beta_5^{L0} L_0 \times box_{resp} + \beta_5^{L1} L_1 \times box_{resp} + \beta_6 box_{items} + \beta_L Left + \beta_R Right$$

where L_0 and $L1$ correspond to the different levels of interest (e.g., high TCC and low TCC respectively) and box_{Si}, box_{resp}, box_{items} correspond to the boxcar functions of the solving, response and items stages, respectively. $Left$ and $Right$ correspond to the button pressed by the participant.

Group level analyses were performed using mixed effects multilevel modeling (Chen et al. 2012). All whole-brain analysis results are reported with a clusterwise threshold of $p < 0.05$ corrected for multiple comparisons across the whole brain, using an uncorrected voxelwise threshold of $p < 0.001$.

ROI specification

We were particularly interested in how control and subjective beliefs of cognitive demand and reliability were involved in complex problem-solving. To study these dynamic processes we selected three regions of interest (ROIs) that have been implicated in the processes of interest. Firstly, we included in our analysis the CON (dACC and AI) due to its proposed involvement in the allocation of control (Shenhav, Botvinick, and J. D. Cohen 2013; Dosenbach, Visscher, et al. 2006; Silvetti et al. 2018; Vassena, Holroyd, and Alexander 2017; Holroyd and Yeung 2012; Alexander and Brown 2011) and uncertainty encoding (Neta, Steven M Nelson, and Petersen 2017; Neta, Schlaggar, and Petersen 2014; Bossaerts 2018), which we conjectured would be highly related to encoding of reliability. Secondly, we included a region that has been involved in moment-to-moment processing operations during problem-solving. We expected the knapsack task to engage processing units associated with number processing and mathematical calculations. Therefore, we selected a region that has been widely connected to 'processing' in mathematical problem-solving, the right IPS (Matejko and Ansari 2018; Brannon 2006; Arsalidou and M. J. Taylor 2011).

The three ROIs were selected from the clusters found when contrasting high and low TCC in the last boxcar during the solving stage (period S4). We chose the contrast for the fourth boxcar for a few reasons. We expected that during this last period of the solving stage we would be able to see a marked differentiation in the cognitive demand between instances with high and low TCC. We expected instances with low TCC to require less computational time and thus, we hypothesized that, on average, participants would be still making calculations during the period S4 for high TCC instances, but not for low TCC instances. This was further indicated by a parallel pilot study that found that participants spent on average 17.9s solving an instance with low TCC and 21.2s on those with high TCC (period S3 ends at 19.5s of solving stage). Importantly, we believed that these differences in cognitive demand would be reflected, as well, in a differentiation in the control activity in the system. Therefore, we expected that significant clusters found in this period

would capture differences in neural markers associated to control. Critically, we expected the monitoring of control variables such as expected performance would differ between types of instances. For instance, we expected the subjective markers of performance would converge to actual performance levels in the late stages of the solving stage (Franco, Yadav, et al. 2020; Fig 5.2.2), which would imply higher subjective beliefs of expected performance for low TCC. Additionally, we expected that this contrast would allow us to control for task-set signals (Dosenbach, Fair, Miezin, et al. 2007). We conjectured that the task-set signals would be maintained during the whole solution-stage, so the proposed contrast would not capture task-set signals encoding goals nor the underlying structure of the task.

Among the significant clusters found around the right IPS, we chose the IPS (AG) cluster (peak: x=32, y =-65, z=47) because of its overlap with the regions that were found to be associated with mathematical calculations in the meta analysis by Arsalidou and M. J. Taylor 2011.

ROI temporal dynamics

We explored the dynamics in these ROIs by fitting generalized linear models (GLM) using AFNI (Cox 1996). Analogous to the whole brain GLM analysis (i.e., boxcar analysis), we spatially smoothed the signal and censored outliers from the regression. In this case, in contrast to the whole brain analysis GLMs, we modeled the trial time using a Finite Impulse Response (FIR) approach, in which each trial was modeled using 17 simple basis functions (tents; Fig 5.2.1). This approach allowed us to take advantage of the short TRs (0.8s) used for the functional acquisition sequence, which were possible due to the ultra-high-field MRI used in the experiment. Modeling the BOLD signal using FIR allowed us to obtain 17 beta estimates β_{FIR} for each voxel for each of the conditions considered. Note that these estimates model the hemodynamic response directly and, therefore, they do not factor in the lag of the BOLD signal. In order to link each β_{FIR} to a time in the trial, we assumed a lag of 5 seconds in the hemodynamic response.

We obtained a 2x2 β_{FIR}-estimates for the factors TCC (high and low) and satisfiability (satisfiable and unsatisfiable). We explored the dynamics of each ROI by estimating the average β_{FIR} over all of the voxels from each ROI for each condition. The ROI signal aggregation was performed using python 3.7 and the nilearn library.

Connectivity analysis

Connectivity analysis was performed over the three ROIs. To remove non-neural sources from the neural signal, the motion parameters were regressed out before extracting the relevant ROI signals. We then performed connectivity analysis using two separate approaches.

Psychophysiological interaction (PPI) We performed generalized PPI analyses using AFNI. We ran two separate regressions for each ROI; one for satisfiability and one for TCC. Each PPI regression was estimated according to the following:

$$\begin{aligned} y =& \beta_0 + \beta_1 S_{ROI} + \beta_3 L_0 \times box_{PPI1} + \beta_4 L_1 \times box_{PPI1} + \beta_5 L_0 \times box_{PPI2} + \\ & \beta_6 L_1 \times box_{PPI2} + \beta_7 L_0 \times box_{PPI1} \times S_{ROI} + \beta_8 L_1 \times box_{PPI1} \times S_{ROI} + \\ & \beta_9 L_0 \times box_{PPI2} \times S_{ROI} + \beta_{10} L_1 \times box_{PPI2} \times S_{ROI} \end{aligned}$$

where L_0 corresponds to low TCC (or satisfiable) condition and L_1 corresponds to high TCC (or unsatisfiable) condition. S_{ROI} is the neural signal of the seed region and box_i corresponds to a boxcar function that separates the items and solving stages, together, into two boxcar functions (PPI-1 and PPI-2) of the same duration (12.5s each; Fig 5.2.7). Note that these boxcar functions are different in duration to the ones used for the boxcar GLM analysis. The contrasts of interest (β_7, β_8, β_9 and β_{10}) captured the PPI effects; that is, the task-dependent connectivity to the ROIs for each of the two periods considered. Additionally, we tested whether there were regions that showed a differential connectivity to an ROI between conditions (i.e., high vs. low TCC, unsatisfiable vs. satisfiable). Explicitly, we performed group level analysis using mixed effects multilevel modeling (Chen et al. 2012) on the contrasts corresponding to $L_1 - L_0$ ($\beta_8 - \beta_7$ and $\beta_{10} - \beta_9$). Results are reported with a clusterwise threshold of $p < 0.05$ corrected for multiple comparisons across the whole brain, using an uncorrected voxelwise threshold of $p < 0.001$.

It is worth noting that the interaction between box-car functions and the seed region ($box \times S$) was estimated via deconvolution. That is, the BOLD time series of each seed region was deconvolved with a canonical HRF (AFNI: *BLOCK(0.1,1)*) and then multiplied with the psychological boxcar function. This was convolved back with the same HRF to form a predicted PPI time series at the hemodynamic response level (BOLD), at which the regression takes place.

Granger causality Additionally, we performed Granger Causality (GC) analysis on the three ROIs. To do this, we first fitted a DCM to the BOLD time series of these ROIs and estimated GC on the residuals of the model. This was done to ensure that the DCM captured all the task-relevant events and controls not strictly related to the internal solving process itself (e.g., onset of decision screen). We report the exact specification of the DCM in Appendix B.2. We then extracted the residual series of the DCM model for each region. We refrained from deconvolving the BOLD residuals (in accordance with Seth, Chorley, and Barnett 2013) because deconvolution is a smoothing operation that introduces spurious lead-lag relationships.

GC emerges when lagged outcomes of a variable *correlate* significantly with values of another variable. As such, GC is closely linked to *cross-autocorrelations*. Typically, GC is analyzed in the context of a Vector Auto Regression (VAR), i.e., a model whereby a vector of outcomes is driven by a finite number of lags of itself. GC emerges when the presence of lags of one variable significantly improves the fit (maximum likelihood value) of another variable. If this is the case, the former "Granger causes" (GCs) the latter. We ran a VAR on the time series augmented with the time series during the solving stage only, and determined incremental GC of one series on another during problem-solving.

In order to reach a GC statistic at the group level we carried out the following procedure. We first ran a VAR for each subject. Each subject's VAR maximum lag was determined by comparing AIC (Akaike Information Criterium) for lags up to 10. From each regression we extracted 5 GC statistics for each ROI: 2 GCs from lagged time series of each of the other two ROIs and 3 GCs (one for each ROI) from the lagged time series of the solving stage. This process generates 15 GC statistics per subject. To correct for multiple comparisons among these we performed standard Bonferroni correction.

To determine statistical significance at the group level, a standard binomial test

was then employed to determine the significance of the frequency of rejections (of no GC) across the 20 participants. A p level of was 0.05 used. FWE correction was applied using Holm-Bonferroni correction over the 15 tests.[2]

The Matlab method `gctest` was used to implement the Granger Causality estimations.

5.4.10 Data and code availability

The data analysis code and the behavioral data will be made available upon publication at the Open Science Framework (OSF). The software for the knapsack decision task will be made available there as well. The anonymized neuroimaging data will be made available (in BIDS format) upon publication. The software for the knapsack optimization task and mental arithmetic task correspond to those employed by Franco, Yadav, et al. 2020 and are available at the OSF (DOI 10.17605/OSF.IO/T2JV7).

Acknowledgments

The authors thank Rebecca Glarin for her support of the scanning sessions and Scott Kolbe for his guidance on the 7T MRI setup.

This research is supported by a University of Melbourne Graduate Research Scholarship from the Faculty of Business and Economics. Bossaerts acknowledges financial support through a R@MAP Chair from the University of Melbourne. The authors also acknowledge financial support through a Research Development Grant from the Faculty of Business and Economics.

Competing interests

The authors declare no competing interests.

Appendices

Appendix A Complementary tasks

A.1 The knapsack optimization task

In this task, participants were asked to solve a number of instances of the (0-1) knapsack optimization problem (Fig A.1(a)). In each trial, they were shown a set of items with different weights and values as well as a capacity constraint. Participants had to find the subset of items that maximized total value subject to the capacity constraint. This means that while in the knapsack decision task, participants only needed to determine whether a solution existed, in the knapsack optimization task,

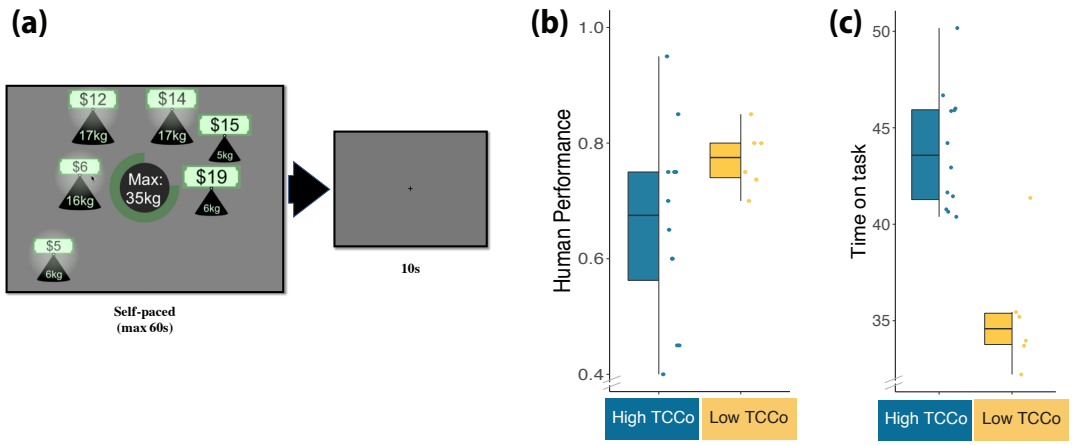
[2]Significance is determined as follows: order p values from small ($k = 1$) to large ($k = 15$); the kth test value is deemed to be significant at the level α if $p(k) \leq \alpha/(m + 1 - k)$ where m is the number of hypotheses to be tested; here: $m = 15$. If $\alpha = 0.05$ then the smallest p should be ≈ 0.0033 for the corresponding test (i.e., the test with smallest p value) to reject.

they also needed to determine the nature of the solutions (i.e., the items in the optimal knapsack).

The task consisted of a single solving stage (60 seconds) and an inter-trial interval (fixation cross for 10 seconds). During the solving stage the items and the capacity constraint were presented in the same way as in the knapsack decision task. Unlike in the decision task, however, there was no target profit and participants were able to add and remove items to/from the knapsack by clicking on the items. An item added to the knapsack was indicated by a halo around it (Fig A.1). Participants could submit their solution before the time limit was reached. If participants did not submit within the time limit, the items selected at the end of the trial were automatically submitted as the solution. Participants were then shown a fixation cross (10 seconds) before the start of the next trial.

Each participant completed 18 trials (2 blocks of 9 trials with a rest period of 60 seconds between blocks). Each trial presented a different instance of the knapsack optimization problem with varying levels of computational complexity. The order of presentation of instances in the task was randomized for each participant.

Figure A.1: **Knapsack optimization task. (a) Experimental design.** Participants were presented with a set of items of different values and weights together with a capacity constraint shown at the center of the screen. The green circle at the center of the screen indicated the time remaining in this stage of the trial. Participants had to find the subset of items with the highest total value subject to the capacity constraint. This stage lasted up to 60 seconds. Participants selected items by clicking on them and had the option of submitting their solution before the time limit was reached. After the time limit was reached or they submitted their solution, a fixation cross was shown for 10 seconds before the next trial started. **(b) TCC_O and human performance.** Human performance corresponds to mean computational performance on each instance. **(c) TCC_O and time-on-task.** Mean time spent before skipping to the response screen. *Each dot represents an instance and is categorized according to its TCC_O. The box-plots represent the median, the interquartile range (IQR) and the whiskers extend to a maximum length of 1.5*IQR.*



For this task we aimed at replicating the results found by Franco, Yadav, et al. 2020. In particular, we expected to find an negative effect of optimization TCC (TCC_O) on performance and a positive effect on time-on-task. TCC_O is defined as the TCC of the decision of determining whether the optimal profit (α_p^*) is attainable

given the capacity constraint.

We employed the same instances of the knapsack optimization problem used in Franco, Yadav, et al. 2020. In their study, 12 instances were selected to have high TCC_O and 6 instances were selected to have low TCC_O. All instances had $N = 6$ items and w_i, v_i, c, p were integers.

We investigated participants' ability to find the optimal solution of an instance. We do this by estimating a metric of *computational performance* that is defined as a binary variable that is equal to 1 if the participant obtained a value equal to the maximum value obtainable in the instance, and 0 otherwise. Mean computational performance was 69.6% (min = 0.18, max = 1, $SD = 0.25$). Participants were allowed to select any set of items, irrespective of the capacity constraint, which implied that they could submit candidate solutions that exceeded the capacity constraint. However, the capacity constraint was only violated in 3.9% of instances.

Additionally, we explored the time-on-task. In contrast to the decision variant, the optimization task was self-paced and, as such, participants were allowed to submit their answer before the time limit (60s) was reached. We recorded the time participants spent in the solving stage before submitting their candidate solution. Participants spent on average 41.0 seconds on an instance (min = 21.0, max = 55.8, $SD = 8.1$).

We first analyzed the effect of trial number on the task. We found that performance did not change throughout the task ($\beta_{0.5} = 0.03$, $HDI_{0.95} = [-0.02, 0.09]$, main effect of trial number on computational performance, GLMM; Table A.1 Model 1), nor did the time-on-task per instance ($\beta_{0.5} = -0.02$, $HDI_{0.95} = [-0.03, 0.22]$, main effect of trial number on time-on-task, CLMM; Table A.1 Model 3). These results suggest, in line with previous results (Franco, Yadav, et al. 2020), that neither experience with the task nor mental fatigue affected the quality and speed of finding the a solution.

We expected that performance in instances with *high* TCC_O (instances whose solutions have a corresponding decision problem with high TCC) would be lower than in instances with *low* TCC_O (instances whose solutions have a corresponding decision problem with low TCC). We, indeed find this effect on both computational performance and time-on-task. Mean computational performance was lower in instances with high TCC_O, relative to those with low TCC_O ($\beta_{0.5} = -0.75$, $HDI_{0.95} = [-1.35, -0.14]$, main effect of TCC_O on performance, GLMM; Fig A.1b; Table A.1 Model 2). Similarly, we found a negative relation between time-on-task and the probability of finding the solution ($\beta_{0.5} = 8.78$, $HDI_{0.95} = [6.45, 10.97]$, main effect of TCC_O on time-on-task, CLMM; Fig A.1c; Table A.1 Model 4). These results replicate those found by Franco, Yadav, et al. 2020.

A.2 Cognitive function

In a previous study we tested participants' performance on five aspects of cognitive function that we considered relevant for the knapsack tasks (Franco, Yadav, et al. 2020). Explicitly, we assessed working memory, episodic memory, strategy use, processing and psychomotor speed, as well as mental arithmetic. We were interested in finding links between these cognitive capacities and the ability to solve the knapsack task. A complex task that would arguably require the deployment of these, more basic, cognitive abilities. Our original study lacked the power to identify reliably

Table A.1: **Computational performance and time-on-task in the knapsack optimization task.** Models on computational performance represent logistic regressions with random intercept effects for participants. Regression parameters relate performance to trial number (1), and optimization typical-case complexity (TCC_O) (2). Models on time-on-task represent censored linear regressions (with random intercept effects for participants) relating time spent on an instance to trial number (3), and optimization typical-case complexity (TCC_O) (4). *Parameter estimates correspond to the median of the posterior distribution ($\beta_{0.5}$) and the 95% HDI credible interval ($HDI_{0.95}$). ELPD denotes the expected log posterior predictive density.*

	Dependent variable			
	Computational performance		Time-on-task	
	(1)	(2)	(3)	(4)
Trial Number	0.03		-0.02	
	[-0.02,0.09]		[-0.26,0.22]	
TCC_O		-0.75		8.78
		[-1.35,-0.14]		[6.45,10.97]
Intercept	0.84	1.61	41.21	35.12
	[0.05,1.7]	[0.74,2.45]	[36.51,45.47]	[30.98,39.98]
Observations	359	359	323	323
ELPD	-184.19	-181.54	-1228.64	-1199.62

correlations between performance in these cognitive tasks and performance in the knapsack tasks.

In this study we tested participants on the same five aspects of cognitive function with the aim of increasing the power of these exploratory tests. For this purpose, we aggregated the data collected in this study with that collected by (Franco, Yadav, et al. 2020) and estimated the same correlations presented in our previous study.

Following the approach by Franco, Yadav, et al. 2020 we administered a set of tasks from the Cambridge Neuropsychological Test Automated Battery (CANTAB; Cognition 2017). Specifically, we asked participants to perform the Paired Associates Learning (PAL), Spatial Working Memory (SWM) and Spatial Span (SSP). Additionally, participants solved a set of mental arithmetic problems (Cappelletti, Butterworth, and Kopelman 2001). Below we describe each of the tests performed:

Paired Associates Learning (PAL) Boxes are displayed on the screen and open one by one in a randomized order to reveal patterns hidden inside. The patterns are then displayed in the middle of the screen, one at a time, and the subject must touch the box where the pattern was originally located.

Spatial Working Memory (SWM) The test begins with colored boxes being shown on the screen. The aim of this test is that, by touching the boxes and using a process of elimination, the subject should find one 'token' in each of the boxes and use them to fill up an empty column on the right hand side of the screen. The

computer will never hide a token in the same colored box, so once a token is found in a box the participant should not return to that box to look for another token.

Spatial Span Task (SSP) White squares briefly change color in a variable sequence. The participant must remember the sequence and then touch the squares in that same order. The sequence length increases through the test. There are up to 3 attempts at each sequence length and the test terminates if all three are failed.

Mental Arithmetic Task Participants were asked to answer a set of 33 mental arithmetic problems. They were given 13 seconds to solve each problem. The task involved addition and division of numbers, as well as questions in which they were asked to round to the nearest integer the result of an addition or division operation.

From performance in these tasks we estimated five metrics of cognitive capacities and estimated their correlation with participant's performance on the knapsack decision and optimization tasks. Results are presented in Table A.2. We found, after correcting for multiple comparisons using Holm-Bonferroni correction, a significant positive effect between performance in the knapsack optimization task and performance in the mental arithmetic task ($\rho = 0.617$ at FWE-corrected $\alpha = 0.05$). Additionally, we found (at FWE-corrected $\alpha = 0.10$) a negative correlation between the *strategy use* metric and performance in the knapsack decision task ($\rho = -0.421$). The SWMS metric encodes the number of times a subject begins a new search pattern from the same box they started with previously in the SWM task. Therefore, a lower score is interpreted as higher strategy use (1 = they always begin the search from the same box). These results suggest that participants that use a planned strategy in SWM perform better in the knapsack decision task.

Table A.2: **Pearson correlations between performance in the knapsack tasks and cognitive abilities.** Performance in the knapsack decision task is characterized by accuracy and in the knapsack optimization task is characterized by computational performance. The cognitive abilities measured used were mental arithmetic, episodic memory (PALFAMS28), working memory (SSPFSL), strategy use (SWMS) and spatial working memory (weighted SWMTE, with errors on easier tasks being weighted more). P-values are shown without multiple comparisons correction. *Note: FWE significance * <0.1; ** <0.05; *** <0.01 is assessed employing Holm-Bonferroni correction.* [1]In the mental arithmetic task $df = 37$.

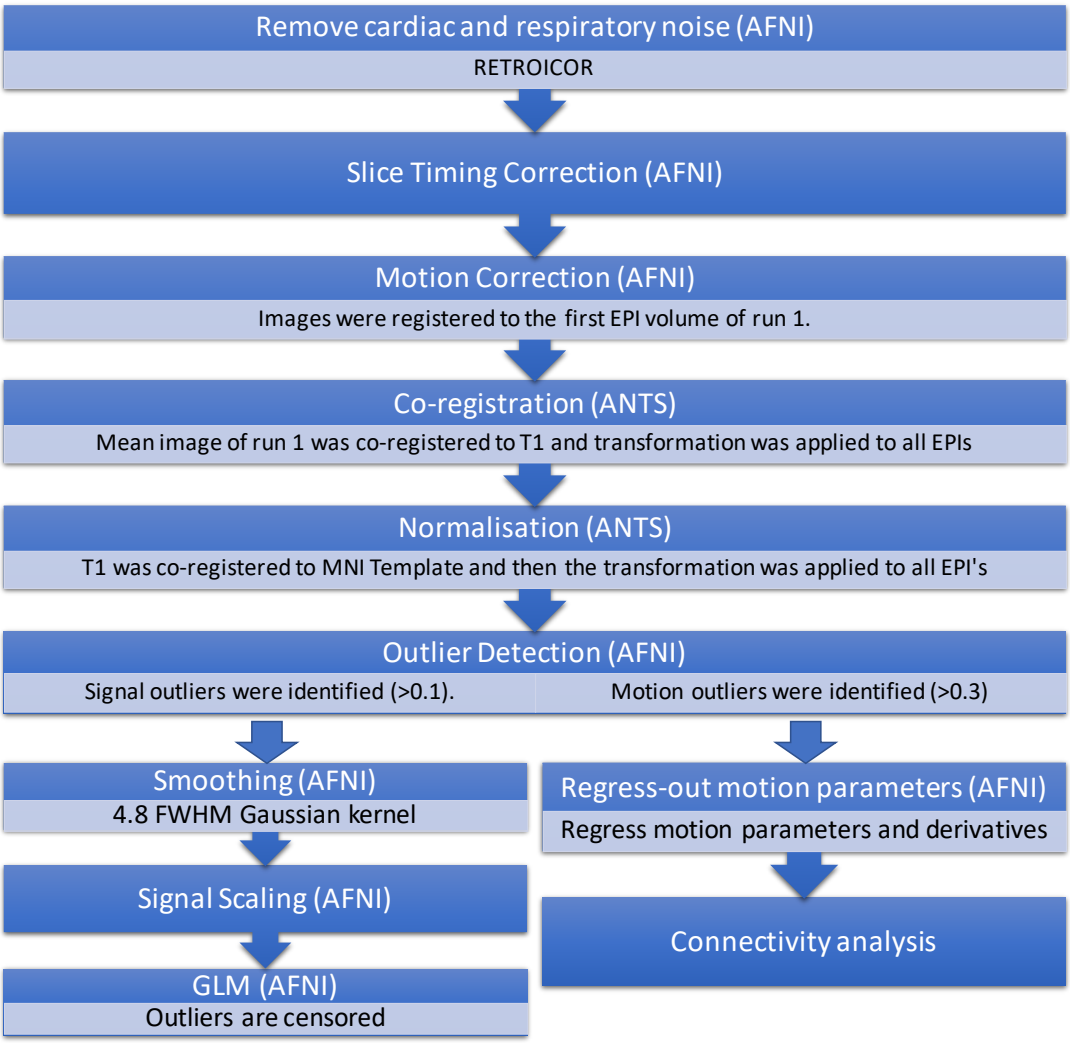
Task	Knapsack decision	Knapsack optimization
Mental arithmetic	0.311 (0.156) p=0.054	0.617*** (0.129) p=0.000
Episodic memory	0.098 (0.161) p=0.549	-0.017 (0.162) p=0.918
Working memory	0.033 (0.162) p=0.839	0.218 (0.158) p=0.177
Strategy use	-0.421* (0.147) p=0.007	-0.336 (0.153) p=0.034
Spatial working memory	-0.360 (0.151) p=0.023	-0.348 (0.152) p=0.028
Degrees of freedom[1]	38	38

Appendix B fMRI analysis

B.1 fMRI preprocessing

Raw images were organized and converted to the relevant format according to the (BIDS) standards. The subsequent preprocessing steps are depicted in Figure B.1 and described below.

Figure B.1: **fMRI data preprocessing pipeline.** Depiction of the preprocessing steps used prior to the statistical analyses performed on the functional data. The preprocessing steps, up to outlier detection, are shared across all types of analyses. Afterwards, preprocessing steps differ between GLMs and functional connectivity models.



Pulse and cardiac noise were regressed out from the functional scans using RETROICOR. These were then slice-time corrected and the volumes were motion-corrected by registering to the first volume of the first functional run. The anatomical (T1) image was down-sampled to the functional EPI resolution ($1.6mm^3$) and the mean BOLD volume of the first run was co-registered to the down-sampled anatomical scan. This transformation was applied to all of the BOLD volumes. Afterwards, each participant's anatomical scan was used for calculation of transformation pa-

rameters to normalize the functional images into the Montreal Neurological Institute (MNI) space.

Whole-brain analyses were performed by fitting generalized linear models (GLM) using AFNI (Cox 1996). Before the regressions were implemented, we spatially smoothed the functional volumes with a 4.8mm FWHM Gaussian kernel. Each voxel's signal was then scaled (per run) to have the same mean (100). Additionally, volumes with motion or signal outliers were censored from each of the regressions. Regressions were performed using the 3dREMLfit algorithm in AFNI. Group level statistical tests were performed using mixed effects multilevel modeling (Chen et al. 2012).

In the connectivity analyses the volumes were not smoothed, but motion parameters were regressed out before extracting the relevant ROI signals. Whitened (ARMA(1,1)) residuals were used in the subsequent analyses.

B.2 DCM specification

Additional to the preprocessing steps presented in the previous section we fitted a dynamic causal model (DCM) before Granger causality (GC) analysis. This was done in order to remove signals of no interest related to perceptual processes related to screen and stage changes. The model was fit using the SPM12 software (https://www.fil.ion.ucl.ac.uk/spm/software/spm12/) and the residuals of the resulting model were then used to fit the VAR model and test for GC (see section 5.4.9). In this section we describe the DCM used.

Let z denote a vector of neural activity in 3 regions, indexed i $(= rAI, rAG, dACC)$.

Let v_j denote conditions; they reflect stages in the task. The variable is a dummy variable that takes the value of 1 when the j condition is ON the screen:

- $j = 1$: items stage and solving stage (25s).
- $j = 2$: response stage (2s).

Additionally, let o_j denote onsets of conditions; that is, when a stage becomes visible on the screen.

We follow SPM's notation to describe the model employing three different types of matrices. Matrix A specifies the baseline effective connectivity. Matrix $B^{(j)}$ denotes the modulation of effective connectivity due to experimental condition j. Finally, $C^{(j)}$ captures the change of the neural response due to the onset of condition j.

The DCM fit is described by the following three equations; one for each ROI:

For rAG:

$$\begin{aligned}\frac{dz_{rAG}}{dt} = &-0.5 \exp\left(A_{rAG} + B^{(1)}_{rAG}(u_1)\right) z_{rAG} + \\ &\left(B^{(1)}_{dACC \to rAG}(u_1)\right) z_{dACC} + \left(B^{(1)}_{rAI \to rAG}(u_1)\right) z_{rAI} + \\ &+ C^{(1)}_{rAG} o_1;\end{aligned}$$

For rAI:

$$\frac{dz_{rAI}}{dt} = -0.5 \exp\left(A_{rAI} + B^{(1)}_{rAI}(u_1)\right) z_{rAI} + \left(B^{(1)}_{dACC \to rAI}(u_1)\right) z_{dACC} + \left(B^{(1)}_{rAG \to rAI}(u_1)\right) z_{rAG} + + C^{(2)}_{rAI} o_2;$$

As to dACC:

$$\frac{dz_{dACC}}{dt} = -0.5 \exp\left(A_{dACC} + B^{(1)}_{dACC}(u_1)\right) z_{dACC} + \left(B^{(1)}_{rAG \to dACC}(u_1)\right) z_{rAG} + \left(B^{(1)}_{rAI \to dACC}(u_1)\right) z_{rAI} + + C^{(2)}_{dACC} o_2;$$

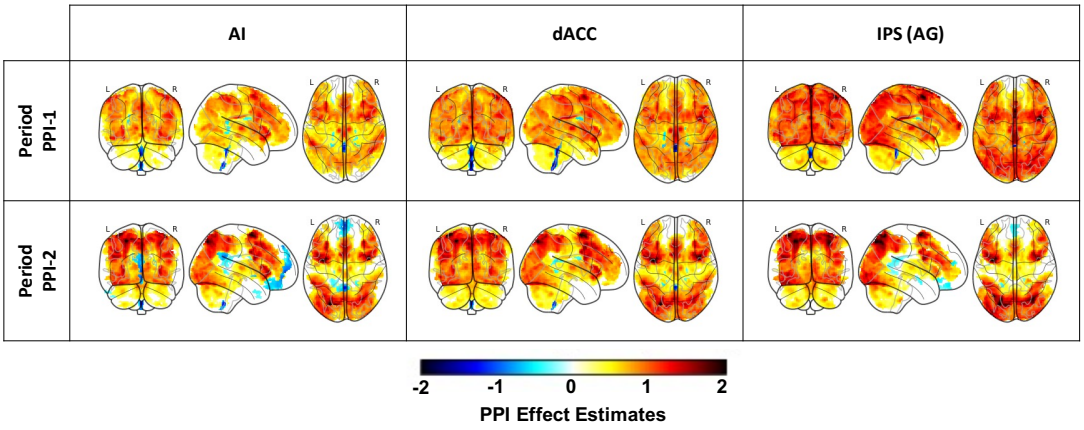
It is worth noting the asymmetries between regions in our specification. These are found in the burst of activity in the model (C matrix). Specifically, we expected the AG to be a processing unit with activity starting quickly from the items stage in the task; this is reflected in the $C_{rAG,1} o_1$ term in the rAG equation. In contrast, we expected the AI and dACC to present burst activity related to control and monitoring signals at the moment the solving stage ends (i.e., $C_{rAI,2} o_2$ and $C_{dACC,2} o_2$). Besides this asymmetry, the model allows for a symmetric inter-connectivity between ROIs during the items and solving stages of the task.

Appendix C Tables and Figures

Table C.1: **Human performance in the knapsack decision task.** Logistic regressions, with random intercept effects for participants, relating the accuracy in an instance with trial number (1), typical-case complexity (TCC) (2), instance complexity (IC) (3), the number of witnesses (4), satisfiability (5), as well as TCC and satisfiability (6). *Parameter estimates correspond to the median of the posterior distribution ($\beta_{0.5}$) and the 95% HDI credible interval ($HDI_{0.95}$). ELPD denotes the expected log posterior predictive density.*

	Dependent variable: Human performance					
	(1)	(2)	(3)	(4)	(5)	(6)
Trial Number	0.01 [0.00, 0.02]					
TCC		-1.10 [-1.44, -0.79]				-1.23 [-1.66, -0.79]
IC			6.54 [4.67, 8.29]			
No. of Witnesses				0.20 [0.12, 0.28]		
Satisfiability					0.02 [-0.30, 0.30]	-0.14 [-0.61, 0.37]
TCC:Satisfiability						0.26 [-0.37, 0.9]
Intercept	1.24 [0.83,1.66]	2.05 [1.61,2.52]	0.60 [0.14,1.03]	0.65 [-0.02,1.33]	1.41 [1.00,1.81]	2.13 [1.63,2.67]
Observations	1120	1120	1120	560	1120	1120
ELPD	-546.77	-523.90	-516.67	-237.33	-548.05	-525.37

Figure C.1: **PPI supplementary results.** The effect of the task on the connectivity to each of the three seed regions used for the analysis (dACC, rAG and rAI). Each column shows the PPI effect for a different seed region. Each row displays the period of the task considered. Activation patterns represent the significant PPI-effect estimates in instances with low TCC. The effect for instances with high TCC is not displayed. The only significant differences between high and low TCC conditions are presented in figure 5.2.7a. *Significant cluster-wise FWE-corrected ($p < 0.05$) clusters (with an uncorrected threshold of $p < 0.001$) are presented.*



References

Aben, Bart et al. (2020). "Cognitive effort modulates connectivity between dorsal anterior cingulate cortex and task-relevant cortical areas". In: *Journal of Neuroscience* 40.19, pp. 3838–3848. ISSN: 15292401. DOI: 10.1523/JNEUROSCI.2948-19.2020. URL: https://doi.org/10.1523/JNEUROSCI.2948-19.2020.

Achlioptas, Dimitris, Assaf Naor, and Yuval Peres (2005). "Rigorous Location of Phase Transitions in Hard Optimization Problems". In: *Nature* 435.7043, pp. 759–64. ISSN: 1476-4687. DOI: 10.1038/nature03602. URL: http://www.ncbi.nlm.nih.gov/pubmed/15944693.

Acuña, Daniel E. and Víctor Parada (July 2010). "People efficiently explore the solution space of the computationally intractable traveling salesman problem to find near-optimal tours". In: *PLoS ONE* 5.7. Ed. by Edward Vul, e11685. ISSN: 19326203. DOI: 10.1371/journal.pone.0011685. URL: https://dx.plos.org/10.1371/journal.pone.0011685.

Alexander, William H. and Joshua W. Brown (Oct. 2011). "Medial prefrontal cortex as an action-outcome predictor". In: *Nature Neuroscience* 14.10, pp. 1338–1344. ISSN: 10976256. DOI: 10.1038/nn.2921. URL: https://www.nature.com/articles/nn.2921.

Arsalidou, Marie and Margot J Taylor (2011). "Is 2+2=4? Meta-analyses of brain areas needed for numbers and calculations". In: *NeuroImage* 54.3, pp. 2382–2393. ISSN: 10538119. DOI: 10.1016/j.neuroimage.2010.10.009.

Assem, Moataz et al. (2020). "A Domain-General Cognitive Core Defined in Multimodally Parcellated Human Cortex". In: *Cerebral Cortex* 30.8, pp. 4361–4380. ISSN: 14602199. DOI: 10.1093/cercor/bhaa023.

Basso, Demis, Patrizia Silvia Bisiacchi, et al. (June 2001). "Planning times during traveling salesman's problem: Differences between closed head injury and normal subjects". In: *Brain and Cognition* 46.1-2, pp. 38–42. ISSN: 02782626. DOI: 10.1016/S0278-2626(01)80029-4.

Basso, Demis and Chiara Saracini (2020). "Differential involvement of left and right frontoparietal areas in visuospatial planning: An rTMS study". In: *Neuropsychologia* 136, p. 107260. ISSN: 18733514. DOI: 10.1016/j.neuropsychologia.2019.107260. URL: https://doi.org/10.1016/j.neuropsychologia.2019.107260.

Bielak, Allison A.M., Cassandra R Hatt, and Manfred Diehl (2017). "Cognitive Performance in Adults' Daily Lives: Is There a Lab-Life Gap?" In: *Research in Human Development* 14.3, pp. 219–233. ISSN: 15427617. DOI: 10.1080/15427609.2017.1340050.

Billeke, Pablo et al. (2020). "Human Anterior Insula Encodes Performance Feedback and Relays Prediction Error to the Medial Prefrontal Cortex". In: *Cerebral cortex* 30.7, pp. 4011–4025. ISSN: 14602199. DOI: 10.1093/cercor/bhaa017.

Bossaerts, Peter (2018). "Formalizing the function of anterior insula in rapid adaptation". In: *Frontiers in Integrative Neuroscience* 12. ISSN: 16625145. DOI: 10.3389/fnint.2018.00061. URL: www.frontiersin.org.

Bossaerts, Peter and Carsten Murawski (2017). "Computational Complexity and Human Decision-Making". In: *Trends in Cognitive Sciences* 21.12, pp. 917–929. ISSN: 1879307X. DOI: 10.1016/j.tics.2017.09.005.

Bourgin, David et al. (2017). "The Structure of Goal Systems Predicts Human Performance". In: *Proceedigns of the 39th Annual Meeting of the Cognitive Science Society*. Ed. by G Gunzelmann et al. Austin, TX: Cognitive Science Society, pp. 1660–1665.

Brannon, Elizabeth M (2006). "The representation of numerical magnitude". In: *Current Opinion in Neurobiology* 16.2, pp. 222–229. ISSN: 09594388. DOI: 10.1016/j.conb.2006.03.002. URL: www.sciencedirect.com.

Bürkner, Paul-Christian (2017). "brms: An R Package for Bayesian Multilevel Models Using Stan". In: *Journal of Statistical Software* 80.1, pp. 1–28. DOI: 10.18637/jss.v080.i01.

Cappelletti, Marinella, Brian Butterworth, and Michael Kopelman (2001). "Spared numerical abilities in a case of semantic dementia". In: *Neuropsychologia* 39, pp. 1224–1239. URL: www.elsevier.com/locate/neuropsychologia.

Carruthers, Sarah, Michael E J Masson, and Ulrike Stege (2012). "Human Performance on Hard Non-Euclidean Graph Problems: Vertex Cover". In: *The Journal of Problem Solving* 5.1, p. 34. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1142.

Cheeseman, Peter, Bob Kanefsky, and William M Taylor (1991). "Where the Really Hard Problems Are". In: *The 12nd International Joint Conference on Artificial Intelligence*, pp. 331–337. ISBN: 1-55860-160-0. DOI: 10.1.1.97.3555.

Chen, Gang et al. (Mar. 2012). "FMRI group analysis combining effect estimates and their variances". In: *NeuroImage* 60.1, pp. 747–765. ISSN: 10538119. DOI: 10.1016/j.neuroimage.2011.12.060.

Christopoulos, George I. et al. (Oct. 2009). "Neural correlates of value, risk, and risk aversion contributing to decision making under risk". In: *Journal of Neuroscience* 29.40, pp. 12574–12583. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.2614-09.2009. URL: www.jneurosci.org.

Cognition, Cambridge (2017). *CANTAB® [Cognitive assessment software]*. URL: www.cantab.com.

Cox, Robert W. (1996). "AFNI: Software for analysis and visualization of functional magnetic resonance neuroimages". In: *Computers and Biomedical Research* 29.3, pp. 162–173. ISSN: 00104809. DOI: 10.1006/cbmr.1996.0014. URL: https://pubmed.ncbi.nlm.nih.gov/8812068/.

Crittenden, Ben M, Daniel J Mitchell, and John Duncan (2016). "Task encoding across the multiple demand cortex is consistent with a frontoparietal and cingulo-opercular dual networks distinction". In: *Journal of Neuroscience* 36.23, pp. 6147–6155. ISSN: 15292401. DOI: 10.1523/JNEUROSCI.4590-15.2016.

Crottaz-Herbette, S and V Menon (2006). "Where and when the anterior cingulate cortex modulates attentional response: Combined fMRI and ERP evidence". In: *Journal of Cognitive Neuroscience* 18.5, pp. 766–780. ISSN: 0898929X. DOI: 10.1162/jocn.2006.18.5.766. URL: http://psyscope.psy.cmu.edu.

D'Acremont, Mathieu and Peter Bossaerts (2008). "Neurobiological studies of risk assessment: a comparison of expected utility and mean-variance approaches." In: *Cognitive, affective & behavioral neuroscience* 8.4, pp. 363–74. ISSN: 1530-7026. DOI: 10.3758/CABN.8.4.363. URL: http://www.ncbi.nlm.nih.gov/pubmed/19033235.

D'Acremont, Mathieu and Peter Bossaerts (2016). "Neural Mechanisms behind Identification of Leptokurtic Noise and Adaptive Behavioral Response". In: *Cerebral Cortex* 26.4, pp. 1818–1830. ISSN: 14602199. DOI: 10.1093/cercor/bhw013.

d'Acremont, Mathieu, Wolfram Schultz, and Peter Bossaerts (2013). "The human brain Encodes event frequencies while forming subjective beliefs". In: *Journal of Neuroscience* 33.26, pp. 10887–10897. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.5829-12.2013.

De Smedt, Bert, Ian D Holloway, and Daniel Ansari (2011). "Effects of problem size and arithmetic operation on brain activation during calculation in children with varying levels of arithmetical fluency". In: *NeuroImage* 57.3, pp. 771–781. ISSN: 10538119. DOI: 10.1016/j.neuroimage.2010.12.037.

De Visscher, Alice and Marie Pascale Noël (2014). "The detrimental effect of interference in multiplication facts storing: Typical development and individual differences". In: *Journal of Experimental Psychology: General* 143.6, pp. 2380–2400. ISSN: 00963445. DOI: 10.1037/xge0000029.

Donoso, Maël, Anne G.E. Collins, and Etienne Koechlin (2014). "Foundations of human reasoning in the prefrontal cortex". In: *Science* 344.6191, pp. 1481–1486. ISSN: 10959203. DOI: 10.1126/science.1252254.

Dosenbach, Nico U.F., Damien A Fair, Alexander L Cohen, et al. (2008). "A dual-networks architecture of top-down control". In: *Trends in Cognitive Sciences* 12.3, pp. 99–105. ISSN: 13646613. DOI: 10.1016/j.tics.2008.01.001.

Dosenbach, Nico U.F., Damien A Fair, Francis M Miezin, et al. (2007). "Distinct brain networks for adaptive and stable task control in humans". In: *Proceedings of the National Academy of Sciences of the United States of America* 104.26, pp. 11073–11078. ISSN: 00278424. DOI: 10.1073/pnas.0704320104.

Dosenbach, Nico U.F., Kristina M. Visscher, et al. (June 2006). "A Core System for the Implementation of Task Sets". In: *Neuron* 50.5, pp. 799–812. ISSN: 08966273. DOI: 10.1016/j.neuron.2006.04.031.

Dry, Matthew et al. (2006). "Human Performance on Visually Presented Traveling Salesperson Problems with Varying Numbers of Nodes". In: *The Journal of Problem Solving* 1.1. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1004. URL: http://dx.doi.org/10.7771/1932-6246.1004.

Dubis, Joseph W et al. (2016). "Tasks Driven by Perceptual Information Do Not Recruit Sustained BOLD Activity in Cingulo-Opercular Regions". In: *Cerebral Cortex* 26.1, pp. 192–201. ISSN: 14602199. DOI: 10.1093/cercor/bhu187.

Duncan, John (2010). "The multiple-demand (MD) system of the primate brain: mental programs for intelligent behaviour". In: *Trends in Cognitive Sciences* 14.4, pp. 172–179. ISSN: 13646613. DOI: 10.1016/j.tics.2010.01.004. URL: http://dx.doi.org/10.1016/j.tics.2010.01.004.

Duncan, John and Adrian M Owen (2000). *Common regions of the human frontal lobe recruited by diverse cognitive demands*. DOI: 10.1016/S0166-2236(00)01633-7.

Duverne, Sandrine and Etienne Koechlin (2017). "Rewards and Cognitive Control in the Human Prefrontal Cortex". In: *Cerebral Cortex* 27.10, pp. 5024–5039. ISSN: 14602199. DOI: 10.1093/cercor/bhx210. URL: https://academic.oup.com/cercor/article/27/10/5024/4080829.

Fedorenko, Evelina, John Duncan, and Nancy Kanwisher (2013). "Broad domain generality in focal regions of frontal and parietal cortex". In: *Proceedings of the National Academy of Sciences of the United States of America* 110.41, pp. 16616–16621. ISSN: 00278424. DOI: 10.1073/pnas.1315235110.

Fouragnan, Elsa, Chris Retzler, and Marios G Philiastides (2018). "Separate neural representations of prediction error valence and surprise: Evidence from an fMRI meta-analysis". In: *Human Brain Mapping* 39.7, pp. 2887–2906. ISSN: 10970193. DOI: 10.1002/hbm.24047.

Franco, Juan Pablo, Karlo Doroc, et al. (2021). "Task-independent metrics of computational hardness predict performance of human problem-solving". In: *bioRxiv*. DOI: 10.1101/2021.04.25.441300. URL: https://www.biorxiv.org/content/early/2021/04/26/2021.04.25.441300.

Franco, Juan Pablo, Nitin Yadav, et al. (2020). "Structural properties of individual instances predict human effort and performance on an NP-Hard problem". In: *bioRxiv*. DOI: 10.1101/405449. URL: https://www.biorxiv.org/content/early/2020/07/21/405449.

Garain, D.N and Sanjeev Kumar (2018). "Japanese vs Vedic Methods for Multiplication". In: *International Journal of Mathematics Trends and Technology* 54.3, pp. 228–235. ISSN: 2231-5373. DOI: 10.14445/22315373/ijmtt-v54p525. URL: http://www.ijmttjournal.org.

Gecode Team (2006). *Gecode: Generic Constraint Development Environment*. URL: http://www.gecode.org.

Gent, Ian P and Toby Walsh (1996). "The TSP phase transition". In: *Artificial Intelligence* 88.1-2, pp. 349–358. ISSN: 00043702. DOI: 10.1016/S0004-3702(96)00030-6.

Gent, Ian P. et al. (1996). "The constrainedness of search". In: *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 1*. Portland. Oregon, pp. 246–252.

Gigerenzer, Gerd and Wolfgang Gaissmaier (Jan. 2011). "Heuristic decision making". In: *Annual Review of Psychology* 62, pp. 451–482. ISSN: 00664308. DOI: 10.1146/annurev-psych-120709-145346.

Grabner, Roland H et al. (2009). "To retrieve or to calculate? Left angular gyrus mediates the retrieval of arithmetic facts during problem solving". In: *Neuropsychologia* 47.2, pp. 604–608. ISSN: 00283932. DOI: 10.1016/j.neuropsychologia.2008.10.013.

Gratton, C et al. (2017). "Distinct Stages of Moment-to-Moment Processing in the Cinguloopercular and Frontoparietal Networks". In: *Cerebral cortex* 27.3, pp. 2403–2417. ISSN: 14602199. DOI: 10.1093/cercor/bhw092.

Gratton, Gabriele et al. (Mar. 2018). "Dynamics of cognitive control: Theoretical bases, paradigms, and a view for the future". In: *Psychophysiology* 55.3. ISSN: 14698986. DOI: 10.1111/psyp.13016.

Grinband, Jack et al. (2011). "The dorsal medial frontal cortex is sensitive to time on task, not response conflict or error likelihood". In: *NeuroImage* 57.2, pp. 303–311. ISSN: 10538119. DOI: 10.1016/j.neuroimage.2010.12.027.

Guid, Matej and Ivan Bratko (2013). "Search-Based Estimation of Problem Difficulty for Humans". In: *Artificial Intelligence in Education. AIED 2013. Lecture Notes in Computer Science*. Ed. by Lane H.C. et al. Vol. 7926. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-39112-5{_}131.

Han, Suk Won, Hana P. Eaton, and Rencrossed Marois (June 2019). "Functional Fractionation of the Cingulo-opercular Network: Alerting Insula and Updating Cingulate". In: *Cerebral Cortex* 29.6, pp. 2624–2638. ISSN: 14602199. DOI: 10.

1093/cercor/bhy130. URL: https://academic.oup.com/cercor/article/29/6/2624/5025417.

Hanks, Timothy D. and Christopher Summerfield (Jan. 2017). *Perceptual Decision Making in Rodents, Monkeys, and Humans*. DOI: 10.1016/j.neuron.2016.12.003.

Hirtle, Stephen C. and Tommy Gärling (May 1992). "Heuristic rules for sequential spatial decisions". In: *Geoforum* 23.2, pp. 227–238. ISSN: 00167185. DOI: 10.1016/0016-7185(92)90019-Z.

Holroyd, Clay B. and Nick Yeung (Feb. 2012). "Motivation of extended behaviors by anterior cingulate cortex". In: *Trends in Cognitive Sciences* 16.2, pp. 122–128. ISSN: 13646613. DOI: 10.1016/j.tics.2011.12.008.

Kaufmann, Liane et al. (2009). "Developmental dyscalculia: Compensatory mechanisms in left intraparietal regions in response to nonsymbolic magnitudes". In: *Behavioral and Brain Functions* 5. ISSN: 17449081. DOI: 10.1186/1744-9081-5-35. URL: http://www.behavioralandbrainfunctions.com/content/5/1/35.

Koechlin, Etienne (2016). "Prefrontal executive function and adaptive behavior in complex environments". In: *Current Opinion in Neurobiology* 37, pp. 1–6. ISSN: 18736882. DOI: 10.1016/j.conb.2015.11.004. URL: http://dx.doi.org/10.1016/j.conb.2015.11.004.

Kotovsky, K., J. R. Hayes, and H. A. Simon (Apr. 1985). "Why are some problems hard? Evidence from Tower of Hanoi". In: *Cognitive Psychology* 17.2, pp. 248–294. ISSN: 00100285. DOI: 10.1016/0010-0285(85)90009-X.

Krzakala, Florent et al. (June 2006). "Gibbs States and the Set of Solutions of Random Constraint Satisfaction Problems". In: *Proceedings of the National Academy of Sciences of the United States of America* 104.25, pp. 10318–23. ISSN: 0027-8424. DOI: 10.1073/pnas.0703685104.

López-Góngora, Mariana et al. (2015). "Neurophysiological evidence of compensatory brain mechanisms in early-stage multiple sclerosis". In: *PLoS ONE* 10.8, p. 136786. ISSN: 19326203. DOI: 10.1371/journal.pone.0136786.

Lüdecke, Daniel, Mattan S Ben-Shachar, and Dominique Makowski (2020). "Describe and understand your model's parameters". In: *CRAN*. DOI: 10.5281/zenodo.3731932. URL: https://easystats.github.io/parameters.

MacGregor, James N. and Yun Chu (2011). "Human Performance on the Traveling Salesman and Related Problems: A Review". In: *The Journal of Problem Solving* 3.2, p. 1. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1090. URL: http://dx.doi.org/10.7771/1932-6246.1090.

Marek, Scott and Nico U.F. Dosenbach (Jan. 2019). "Control networks of the frontal lobes". In: *Handbook of Clinical Neurology*. Vol. 163. Elsevier B.V., pp. 333–347. DOI: 10.1016/B978-0-12-804281-6.00018-5.

Matejko, Anna A. and Daniel Ansari (Dec. 2018). "Contributions of functional Magnetic Resonance Imaging (fMRI) to the study of numerical cognition". In: *Journal of Numerical Cognition* 4.3, pp. 505–525. ISSN: 2363-8761. DOI: 10.5964/jnc.v4i3.136. URL: https://jnc.psychopen.eu/article/view/136.

Menon, Vinod and Lucina Q. Uddin (2010). "Saliency, switching, attention and control: a network model of insula function." In: *Brain structure & function* 214.5-6, pp. 655–667. ISSN: 18632661. DOI: 10.1007/s00429-010-0262-0.

Murawski, Carsten and Peter Bossaerts (2016). "How Humans Solve Complex Problems: The Case of the Knapsack Problem". In: *Nature (Scientific Reports)* 6.34851. ISSN: 2045-2322. DOI: 10.1038/srep34851.

Nelson, Steven M. et al. (June 2010). "Role of the anterior insula in task-level control and focal attention". In: *Brain Structure and Function* 214.5-6, pp. 669–680. ISSN: 1863-2653. DOI: 10.1007/s00429-010-0260-2. URL: http://link.springer.com/10.1007/s00429-010-0260-2.

Neta, Maital, Steven M Nelson, and Steven E Petersen (2017). "Dorsal Anterior Cingulate, Medial Superior Frontal Cortex, and Anterior Insula Show Performance Reporting-Related Late Task Control Signals". In: *Cerebral cortex* 27.3, pp. 2154–2165. ISSN: 14602199. DOI: 10.1093/cercor/bhw053.

Neta, Maital, Bradley L Schlaggar, and Steven E Petersen (2014). "Separable responses to error, ambiguity, and reaction time in cingulo-opercular task control regions". In: *NeuroImage* 99, pp. 59–68. ISSN: 10959572. DOI: 10.1016/j.neuroimage.2014.05.053. URL: http://dx.doi.org/10.1016/j.neuroimage.2014.05.053.

Newell, Ben R., Nicola J. Weston, and David R. Shanks (May 2003). "Empirical tests of a fast-and-frugal heuristic: Not everyone "takes-the-best"". In: *Organizational Behavior and Human Decision Processes* 91.1, pp. 82–96. ISSN: 07495978. DOI: 10.1016/S0749-5978(02)00525-3.

Nomura, Emi M et al. (2010). "Double dissociation of two cognitive control networks in patients with focal brain lesions". In: *Proceedings of the National Academy of Sciences of the United States of America* 107.26, pp. 12017–12022. ISSN: 00278424. DOI: 10.1073/pnas.1002431107. URL: www.pnas.org/cgi/doi/10.1073/pnas.1002431107.

O'Neill, Martin and Wolfram Schultz (Oct. 2013). "Risk prediction error coding in orbitofrontal neurons". In: *Journal of Neuroscience* 33.40, pp. 15810–15814. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.4236-12.2013.

Ohlsson, Stellan (2012). "The Problems with Problem Solving: Reflections on the Rise, Current Status, and Possible Future of a Cognitive Research Paradigm 1". In: *The Journal of Problem Solving* 5.1. DOI: 10.7771/1932-6246.1144. URL: http://dx.doi.org/10.7771/1932-6246.1144.

Payne, John W., James R. Bettman, and Eric J. Johnson (1993). *The Adaptive Decision Maker*. DOI: 10.1017/cbo9781139173933.

Percus, Allon, Gabriel Istrate, and Cristopher Moore (2006). *Computational Complexity and Statistical Physics*. Oxford University Press, p. 384. ISBN: 9780199760565.

Ploran, Elisabeth J et al. (2011). "High quality but limited quantity perceptual evidence produces neural accumulation in frontal and parietal cortex". In: *Cerebral Cortex* 21.11, pp. 2650–2662. ISSN: 10473211. DOI: 10.1093/cercor/bhr055. URL: https://academic.oup.com/cercor/article/21/11/2650/278744.

Power, Jonathan D and Steven E Petersen (2013). *Control-related systems in the human brain*. DOI: 10.1016/j.conb.2012.12.009. URL: http://dx.doi.org/10.1016/j.conb.2012.12.009.

Power, Jonathan D. et al. (Nov. 2011). "Functional Network Organization of the Human Brain". In: *Neuron* 72.4, pp. 665–678. ISSN: 08966273. DOI: 10.1016/j.neuron.2011.09.006.

Preuschoff, Kerstin, Peter Bossaerts, and Steven R Quartz (2006). "Neural Differentiation of Expected Reward and Risk in Human Subcortical Structures". In: *Neuron* 51.3, pp. 381–390. ISSN: 08966273. DOI: 10.1016/j.neuron.2006.06.024.

Sadaghiani, Sepideh and Mark D'Esposito (2015). "Functional characterization of the cingulo-opercular network in the maintenance of tonic alertness". In: *Cerebral Cortex* 25.9, pp. 2763–2773. ISSN: 14602199. DOI: 10.1093/cercor/bhu072. URL: https://academic.oup.com/cercor/article/25/9/2763/2926085.

Sahni, Sartaj and Sartaj (Jan. 1975). "Approximate Algorithms for the 0/1 Knapsack Problem". In: *Journal of the ACM* 22.1, pp. 115–124. ISSN: 00045411. DOI: 10.1145/321864.321873.

Sala-Llonch, Roser, David Bartrés-Faz, and Carme Junqué (2015). "Reorganization of brain networks in aging: a review of functional connectivity studies". In: *Frontiers in Psychology* 6. ISSN: 1664-1078. DOI: 10.3389/fpsyg.2015.00663. URL: www.frontiersin.org.

Seeley, William W. et al. (Feb. 2007). "Dissociable intrinsic connectivity networks for salience processing and executive control". In: *Journal of Neuroscience* 27.9, pp. 2349–2356. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.5587-06.2007.

Selman, Bart and Scott Kirkpatrick (Mar. 1996). "Critical behavior in the computational cost of satisfiability testing". In: *Artificial Intelligence* 81.1-2, pp. 273–295. ISSN: 0004-3702. DOI: 10.1016/0004-3702(95)00056-9.

Sestieri, Carlo et al. (2014). "Domain-general signals in the cingulo-opercular network for visuospatial attention and episodic memory". In: *Journal of Cognitive Neuroscience* 26.3, pp. 551–568. ISSN: 15308898. DOI: 10.1162/jocn{_}a{_}00504.

Seth, Anil K., Paul Chorley, and Lionel C. Barnett (Jan. 2013). "Granger causality analysis of fMRI BOLD signals is invariant to hemodynamic convolution but not downsampling". In: *NeuroImage* 65, pp. 540–555. ISSN: 10538119. DOI: 10.1016/j.neuroimage.2012.09.049.

Shenhav, Amitai, Matthew M. Botvinick, and Jonathan D. Cohen (2013). "The expected value of control: An integrative theory of anterior cingulate cortex function". In: *Neuron* 79.2, pp. 217–240. ISSN: 08966273. DOI: 10.1016/j.neuron.2013.07.007. URL: http://dx.doi.org/10.1016/j.neuron.2013.07.007.

Shepard, Roger N. and Jacqueline Metzler (Feb. 1971). "Mental rotation of three-dimensional objects". In: *Science* 171.3972, pp. 701–703. ISSN: 00368075. DOI: 10.1126/science.171.3972.701.

Siegler, Robert S., Karen E. Adolph, and Patrick Lemaire (1996). "Strategy choices actoss the lifespan". In: *Implicit memory and metacognition*. Erlbaum Press, pp. 79–121.

Silvetti, Massimo et al. (Aug. 2018). "Dorsal anterior cingulate-brainstem ensemble as a reinforcement meta-learner". In: *PLoS Computational Biology* 14.8, e1006370. ISSN: 15537358. DOI: 10.1371/journal.pcbi.1006370.

Simon, Herbert A (1956). "Rational choice and the structure of the environment". In: *Psychological Review* 63.2, pp. 129–138. ISSN: 0033295X. DOI: 10.1037/h0042769.

Stazyk, Edmund H., Mark H. Ashcraft, and Mary S. Hamann (1982). "A network approach to mental multiplication". In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* 8.4, pp. 320–335. ISSN: 02787393. DOI: 10.1037/0278-7393.8.4.320.

Van Opheusden, Bas and Wei Ji Ma (2019). *Tasks for aligning human and machine planning*. DOI: 10.1016/j.cobeha.2019.07.002. URL: https://doi.org/10.1016/j.cobeha.2019.07.002.

Van Rooij, Iris et al. (Apr. 2019). *Cognition and Intractability*. Cambridge University Press. DOI: 10.1017/9781107358331.

Vassena, Eliana, Clay B. Holroyd, and William H. Alexander (2017). "Computational models of anterior cingulate cortex: At the crossroads between prediction and effort". In: *Frontiers in Neuroscience* 11.JUN, pp. 1–9. ISSN: 1662453X. DOI: 10.3389/fnins.2017.00316.

Verguts, Tom, Eliana Vassena, and Massimo Silvetti (Mar. 2015). "Adaptive effort investment in cognitive and physical tasks: A neurocomputational model". In: *Frontiers in Behavioral Neuroscience* 9, p. 57. ISSN: 16625153. DOI: 10.3389/fnbeh.2015.00057.

Westbrook, Andrew and Todd S Braver (2015). "Cognitive effort: A neuroeconomic approach." In: *Cognitive, affective & behavioral neuroscience* 15.2, pp. 395–415. ISSN: 1531-135X.

Wu, Tingting et al. (2019). "Anterior insular cortex is a bottleneck of cognitive control". In: *NeuroImage* 195, pp. 490–504. ISSN: 10959572. DOI: 10.1016/j.neuroimage.2019.02.042. URL: https://doi.org/10.1016/j.neuroimage.2019.02.042.

Yadav, Nitin et al. (2020). "Is Hardness Inherent In Computational Problems? Performance Of Human And Digital Computers On Random Instances Of The 0-1 Knapsack Problem". In: *24th European Conference on Artificial Intelligence (ECAI 2020)*.

Yoo, Seng Bum Michael, Benjamin Yost Hayden, and John M Pearson (2021). *Continuous decisions*. DOI: 10.1098/rstb.2019.0664. URL: https://doi.org/10.1098/rstb.2019.0664.

Chapter 6

General Discussion

In this manuscript, I adapted and tested a theoretical framework for the study of computational hardness in human cognition. Using insights from computing theory, I introduced a set of metrics that characterize the intrinsic computational hardness of instances of problems in a generic way. I then tested their applicability to the study of human computation at the behavioral and neural level in a set of human laboratory experiments.

In chapter 2, I adapted the metrics commonly used in the study on computational hardness of random ensembles in computer science to the study of cognition. In particular, I defined two metrics: typical-case complexity (TCC) and instance complexity (IC). These metrics characterize the hardness of instances of decision problems. While TCC captures hardness at the level of a random ensemble of instances, IC characterizes hardness of an individual instance.

In chapter 3, *we* explored the effect of these metrics on human performance in the knapsack problem. We found that these metrics predict decision quality in the knapsack decision task. Moreover, in this chapter we provided a generalization of the TCC metric to optimization problems (TCC_O) and showed that this metric of computational hardness predicts performance and time-on-task in the knapsack optimization task, too.

In chapter 4, *we* showed that TCC and IC are indeed generic metrics of computational hardness. Specifically, we extended and compared our previous results for the knapsack problem to two other canonical computational problems: the traveling salesperson problem and the Boolean satisfiability problem (3SAT). We found that the proposed metrics predict human performance and time-on-task across different tasks in a similar way.

Finally, in chapter 5, *we* applied the framework to the study of the neural processes associated with complex problem-solving. Specifically, we conducted an experiment in which participants solved the knapsack decision task while undergoing ultra-high field fMRI and explored the neural correlates of computational complexity. We showed that the set of brain regions where activity is correlated with computational hardness overlaps with the multiple demand system (MDS). Moreover, we provided evidence that the study of intrinsic properties of intractable problems can be employed to characterize neural invariants of human problem-solving.

The work presented here introduces a novel theoretical framework for the study of cognition. The corresponding experimental results provide support for the premise that a new level of analysis is warranted in the study of cognition: *instance-level*

complexity, that is, a level of analysis of computational hardness that captures the generic intrinsic hardness of instances of problems in a way that is task- and strategy-independent. Instance-level complexity can delineate the boundaries of human computation by characterizing the effect of computational hardness of a task on human behavior. This can then be used to explicitly study the way people's computations adapt to the hardness of an instance of a task. This would provide an avenue for refining algorithmic-level theories such as the heuristics program, where it is still an open question how people select which heuristic to use for any given instance of a task. Moreover, it can shed light on the dynamics of neural processes of problem-solving and cognitive resource allocation.

6.1 Future directions

In this thesis, I put forward a research program that aims at characterizing instance-level complexity and its effects on human computation. However, the work presented here is but one milestone in this program that opens a plethora of questions and possibilities. In what follows, I present several of these potential directions for future research.

6.1.1 Dimensions of computational hardness

In this thesis, I investigated one dimension of computational hardness related to the constrainedness parameter. This dimension has been investigated in computer science, where it was found to capture a source of hardness that makes instances with particular values of constrainedness hard for algorithms (and heuristics) to solve. This analysis, however, is performed for fixed values of instance size. The latter is, undeniably, another dimension of the structural properties of instances that affect human problem-solving (Hirtle and Gärling 1992; Dry, Lee, et al. 2006; Stazyk, Ashcraft, and Hamann 1982; De Visscher and Noël 2014; van Opheusden and Ma 2019). Future research should study the interaction between these two source of complexity in order to characterize how constrainedness and instance size jointly affect human computation.

Importantly, the framework put forward here is based on a line of research in which the the computational requirements commonly investigated are related to time complexity. That is, instances are generally categorized as hard depending on the number of computational steps (or time) needed by algorithms to solve an instance. Nonetheless, there exists another dimension which might be particularly relevant for research in cognition: memory requirements. It is not clear how memory requirements (i.e., space complexity) are affected by the features of the instance. Future work could study how generic properties of instances affect memory requirements and how these are related to human computation. Note that these requirements might impose even tighter constraints on the algorithms that can be implemented by humans, thus they shed additional light on the characterization of human computable algorithms (Blum and Vempala 2020).

The dimensions of computational hardness discussed thus far have been generic sources of hardness. Their generality has several advantages to which I have referred to extensively (section 1.1). However, there might also be sources of hardness that stem from the idiosyncratic structure of a problem or from the features of a single

strategy. For instance, in chapter 4, we found that human performance in the 3SAT was distinctively affected by the satisfiability of the instance. Further work would be needed in order to reach a proper characterization of the multidimensional construct of computational hardness. This would provide an insightful mapping from the features of an instance to the reliability of human problem-solving and the underlying computations.

6.1.2 Approximating optimality: Linking efficiency and reliability

Several computational-level theories of decision-making model humans as agents that optimize (e.g., Samuelson 1938; Nash 1950). However, most of these approaches fail to consider computational feasibility (Bossaerts and Murawski 2017; van Rooij, Blokpoel, et al. 2019). Theories have presented a way forward by proposing that agents approximate the computational-level descriptions of optimality. For instance, it has been proposed that agents satisfice, that is, instead of optimizing, agents target a particular level of a target criterion and aim to reach that level (Simon 1956). This, however, is in itself a computational problem whose computational hardness has not been analyzed systematically. Alternatively, at the algorithmic-level, heuristics have been proposed to model the implementation of computational-level optimization problems. Nevertheless, their ability to approximate the relevant solutions (i.e., their reliability) has not been systemically analyzed either. Nor has their computational hardness been formally studied in order to justify their existence as *fast-and-frugal* effective procedures (Gerd. Gigerenzer and Selten 2001; Otworowska et al. 2018; Rich et al. 2019). Overall, the notions of approximation reliability and its associated computational hardness have been treated rather informally. This raises the question about how to formally approach the investigation of each of these two notions, both independently and, perhaps more importantly, jointly.

Approximation reliability and computational hardness present a trade-off between the computational requirements of reaching an approximate solution and the quality of the approximation. However, the details of this trade-off remain an open question that is currently lacking a principled theoretical framework allowing a thorough investigation of the issue. The approach put forward here could provide such a framework. For example, the metrics presented here could directly characterize the computational hardness of satisficing models, since they can be expressed as computational decision problems. Moreover, this approach can be extended to optimization problems, as shown in chapter 3 for the knapsack problem. Importantly, the generalization proposed can also be employed to characterize the hardness of approximating the optimum by explicitly defining the target approximation reliability. Future work could employ this framework to study the trade-off between selecting an objective and the hardness of reaching said objective. This can shed light on strategy selection and, more specifically, on the question of how adaptation to hardness could drive heuristic selection (Lieder and Griffiths 2017).

6.1.3 Allocation of cognitive resources

The generic framework put forward in this manuscript is particularly valuable for the study of allocation of control during problem-solving given the generality of

this process. The human brain has limited cognitive resources, yet is capable of reusing and reallocating resources in order to successfully solve different problems. Critically, it is able to solve problems that are deemed hard or complex. A true understanding of human cognition would require a generic model of cognitive resource allocation capable of generalizing across several tasks. To this end, several proposals have been put forward in which the allocation of limited resources is modeled as a trade-off between the costs and benefits of expending cognitive resources in a task (e.g., Shenhav, M. M. Botvinick, and J. D. Cohen 2013; Verguts, Vassena, and Silvetti 2015; Westbrook and Braver 2015). The estimation of these costs would require the characterization of cognitive demand.This would probably need to be characterized from generic features of the task such that the agent is able to estimate cognitive demand across tasks. The framework presented here characterizes one component of these costs: computational demand. This dimension of cognitive demand captures the number of computations needed to perform a task based on insights from computational complexity theory. Importantly, this approach is readily generalizable to a plethora of complex problems without the need to assume a particular procedural strategy used to solve a problem. This can inform the study of cognitive resource allocation in order to generate a truly generic model of computational resource allocation. We provided a first exploration on this process by investigating the neural markers of computational hardness and other generic structural properties. Further work is needed in order to assess the generality of these markers across problems and how they are involved in the allocation of cognitive resources during problem-solving.

6.1.4 Fixed-parameter tractability (FPT)

FPT-cognition theory employs problem complexity theory to specify for which models of cognition there could exist strategies that always solve the problem within a reasonable amount of time (van Rooij, Blokpoel, et al. 2019). As such, it is able to characterize the ex-ante feasibility of computational-level theories of cognition. Whilst FPT can be studied mathematically detached from human performance, the framework presented in this manuscript aims at predicting human performance based on intrinsic hardness. Despite this difference, this framework may suggest future avenues of research for FPT.

Our approach identifies a number of features of instances that could help characterizing sources of hardness in FPT. Indeed, the TCC metric presented here stems from research in which an underlying aim has been to identify the structural properties that make instances more likely to be hard. In other words, TCC has been identified as a source of average-case asymptotic complexity. Analogously, the FPT approach provides a theoretical framework to characterize the features of instances that are sources of intractability, that is, sources of worst-case asymptotic complexity (van Rooij, Stege, and Kadlec 2005). Future work could study the connection between worst-case and average-case sources of hardness, and especially, whether TCC is a source of hardness in the worst-case sense.

6.1.5 TCC vs. IC

In this thesis, I introduced two metrics of hardness: TCC and IC. They both stem from the study of random ensembles of instances. However, they capture hardness at two different levels. TCC characterizes the expected computational hardness of an instance sampled from a random ensemble of instances with a given level of constrainedness. IC, on the other hand, captures the computational hardness of a single instance. Together, they facilitate the investigation of human computation at two different levels. Firstly, they can both be used to investigate the effect of computational hardness on human performance and effort in computational tasks. Secondly, TCC can be employed to characterize subjective beliefs of task difficulty. The latter is possible because of the characteristics of TCC, which is a measure that can be estimated without the need to solve an instance, thus making it less computationally intensive. Despite their relation to each other, and to canonical definitions of typical-case complexity in computer science (chapter 2), it remains an open question whether hardness is driven by distance from the satisfiability threshold (α^{sat}), as captured by TCC, or by the distance from the maximum satisfiable constrainedness (α^{*}), as captured by IC. This analysis is problematic because both metrics capture hardness at different levels. One possible way of exploring this is by comparing TCC with IC's corresponding ensemble metric: expected instance complexity (EIC). Further studies could disentangle their effect on human behavior by testing their distinct predictions. Indeed, although both metrics (TCC and EIC) predict a mapping between the satisfiability probability and computational hardness, the details of this mapping differ. Therefore, a more granular mapping between the satisfiability probability and human performance could disentangle their effect.

6.1.6 Landscape analysis

The study of TCC, as it names implies, is an average measure of complexity. This means that it captures the complexity, not of a specific instance, but the expected complexity of a random ensemble. This involves making assumptions about the relevant sampling procedures, which raises a question about the external validity of these procedures. At this point, this question is difficult to answer given that the distribution of real life instances of problems is an open question and very little is known about it (Bogdanov and Trevisan 2006). An alternative approach that avoids the specification of a sampling procedure comes from a line of research in operations research: fitness-landscape analysis. This framework can also be used to characterize the computational requirements of individual instances in a generic way.

The principal goal of landscape analysis has been to determine which algorithm would be best suited for solving a particular problem (Moser, Gheorghita, and Aleti 2017). In order to accomplish this, individual instances of a problem are represented as a landscape in which each location has a corresponding fitness value. The objective is to reach a particular location that maximizes the fitness value based on an *operator function* that defines the feasible moves that can be employed to search the landscape. Importantly, the topology of the fitness landscape has been shown to affect the efficacy of several search algorithms and heuristics (Moser, Gheorghita, and Aleti 2017; Tavares, Pereira, and Costa 2008). In other words, structural properties

of this topology are able to capture intrinsic computational hardness of instances[1].

This representation of a problem is rather generic and can be used to study combinatorial optimization problems such as the knapsack optimization problem (Tavares, Pereira, and Costa 2008) and the traveling salesperson problem (Cicirello 2019; Schiavinotto and Stützle 2007). It is worth noting, however, that this approach has some limitations with regards to its generality and its applicability to the study of human cognition. Firstly, the landscape representation of a problem is not unique (e.g., Cicirello 2019; Tavares, Pereira, and Costa 2008). Critically, the operator function can be described in different ways. This entails a different topology of the landscape representation of the instance, which in turns would imply a different level of computational hardness. Secondly, the topological structure of an instance, and thus its hardness, is computationally intensive to estimate. In order to calculate the topology of an instance, the target location usually needs to be known, which requires solving the problem. Moreover, several other estimations need to be performed in order to characterize landscape metrics of hardness for a particular instance.

In this thesis, I focus my analysis on the study of TCC and related generic metrics of hardness on human cognition. This vantage point allowed me to investigate two things: (1) how computational hardness affects decision quality and (2) how agents adapt to complexity. Landscape analysis might not be suitable for the study of the latter since the estimation of these metrics is computationally intensive. Therefore, it appears unlikely that agents can estimate these metrics to adapt to the hardness of a task. However, landscape analysis can provide an alternative approach to study how people search for solutions and how reliable this can be. Future work should explore whether this framework can be used to capture invariants of human performance and strategy use in a generic way.

* * *

Overall, the work presented here provides a multidisciplinary approach to the study of computational hardness in human cognition. This calls for a closer collaboration between psychologists, neuroscientists and computer scientists. Importantly, the insights from this interdisciplinary approach can inform the development of public policies that aim at minimizing the detrimental effects of computational complexity on decisions. Indeed, in cases where the cognitive demands of a task substantially exceed decision-makers' capacities, there is a need to prevent harm. One way to do so could be AI-powered applications that support people in making complex decisions. Another approach could involve regulatory interventions imposing limits on the complexity of products and/or require product and service providers to generate mechanisms to overcome complexity in the cases where an agent's cognitive capabilities are not sufficient to guarantee a good decision. However, before such mechanisms can be developed and implemented, a well-founded characterization of the complexity of these tasks is needed. In this thesis, I introduced a research program capable of filling this gap.

[1]The reader is referred to Moser, Gheorghita, and Aleti 2017 for a succinct introduction to landscape analysis metrics.

Bibliography

Aaronson, Scott (Feb. 2005). "Guest Column: NP-complete problems and physical reality". In: *ACM SIGACT News* 36.1, p. 30. ISSN: 01635700. DOI: 10.1145/1052796.1052804. URL: http://arxiv.org/abs/quant-ph/0502072.

Aaronson, Scott (2013). "Why Philosophers Should Care About Computational Complexity". In: *Computability: Turing, Gödel, Church, and Beyond*, pp. 261–328.

Aben, Bart et al. (2020). "Cognitive effort modulates connectivity between dorsal anterior cingulate cortex and task-relevant cortical areas". In: *Journal of Neuroscience* 40.19, pp. 3838–3848. ISSN: 15292401. DOI: 10.1523/JNEUROSCI.2948-19.2020. URL: https://doi.org/10.1523/JNEUROSCI.2948-19.2020.

Achlioptas, Dimitris, Assaf Naor, and Yuval Peres (2005). "Rigorous Location of Phase Transitions in Hard Optimization Problems". In: *Nature* 435.7043, pp. 759–64. ISSN: 1476-4687. DOI: 10.1038/nature03602. URL: http://www.ncbi.nlm.nih.gov/pubmed/15944693.

Acuña, Daniel E. and Víctor Parada (July 2010). "People efficiently explore the solution space of the computationally intractable traveling salesman problem to find near-optimal tours". In: *PLoS ONE* 5.7. Ed. by Edward Vul, e11685. ISSN: 19326203. DOI: 10.1371/journal.pone.0011685. URL: https://dx.plos.org/10.1371/journal.pone.0011685.

Alexander, William H. and Joshua W. Brown (Oct. 2011). "Medial prefrontal cortex as an action-outcome predictor". In: *Nature Neuroscience* 14.10, pp. 1338–1344. ISSN: 10976256. DOI: 10.1038/nn.2921. URL: https://www.nature.com/articles/nn.2921.

Ardelius, John and Lenka Zdeborová (2008). "Exhaustive enumeration unveils clustering and freezing in the random 3-satisfiability problem". In: *Physical Review* 78. DOI: 10.1103/PhysRevE.78.040101.

Arora, Sanjeev. and Boaz. Barak (2009). *Computational complexity : a modern approach*. Cambridge University Press, p. 579. ISBN: 0521424267.

Arsalidou, Marie and Margot J Taylor (2011). "Is 2+2=4? Meta-analyses of brain areas needed for numbers and calculations". In: *NeuroImage* 54.3, pp. 2382–2393. ISSN: 10538119. DOI: 10.1016/j.neuroimage.2010.10.009.

Assem, Moataz et al. (2020). "A Domain-General Cognitive Core Defined in Multimodally Parcellated Human Cortex". In: *Cerebral Cortex* 30.8, pp. 4361–4380. ISSN: 14602199. DOI: 10.1093/cercor/bhaa023.

Basso, Demis, Patrizia Silvia Bisiacchi, et al. (June 2001). "Planning times during traveling salesman's problem: Differences between closed head injury and normal subjects". In: *Brain and Cognition* 46.1-2, pp. 38–42. ISSN: 02782626. DOI: 10.1016/S0278-2626(01)80029-4.

Basso, Demis, Martin Lotze, et al. (2006). "The role of prefrontal cortex in visuospatial planning: A repetitive TMS study". In: *Experimental Brain Research* 171.3, pp. 411–415. ISSN: 00144819. DOI: 10.1007/s00221-006-0457-z.

Basso, Demis and Chiara Saracini (2020). "Differential involvement of left and right frontoparietal areas in visuospatial planning: An rTMS study". In: *Neuropsychologia* 136, p. 107260. ISSN: 18733514. DOI: 10.1016/j.neuropsychologia.2019.107260. URL: https://doi.org/10.1016/j.neuropsychologia.2019.107260.

Berg, W Keith et al. (2010). "Deconstructing the tower: Parameters and predictors of problem difficulty on the Tower of London task". In: *Brain and Cognition* 72.3, pp. 472–482. ISSN: 02782626. DOI: 10.1016/j.bandc.2010.01.002.

Blakey, Ed (2011). "Computational complexity in non-Turing models of computation: The what, the why and the how". In: *Electronic Notes in Theoretical Computer Science* 270.1, pp. 17–28. ISSN: 15710661. DOI: 10.1016/j.entcs.2011.01.003.

Blum, Manuel and Santosh Vempala (2020). "The complexity of human computation via a concrete model with an application to passwords". In: *Proceedings of the National Academy of Sciences of the United States of America* 117.17, pp. 9208–9215. ISSN: 10916490. DOI: 10.1073/pnas.1801839117. URL: www.pnas.org/cgi/doi/10.1073/pnas.1801839117.

Bogdanov, Andrej and Luca Trevisan (2006). "Average-Case Complexity". In: *arXiv preprint cs/0606037*. URL: http://arxiv.org/abs/cs/0606037.

Bossaerts, Peter (2018). "Formalizing the function of anterior insula in rapid adaptation". In: *Frontiers in Integrative Neuroscience* 12. ISSN: 16625145. DOI: 10.3389/fnint.2018.00061. URL: www.frontiersin.org.

Bossaerts, Peter and Carsten Murawski (2017). "Computational Complexity and Human Decision-Making". In: *Trends in Cognitive Sciences* 21.12, pp. 917–929. ISSN: 1879307X. DOI: 10.1016/j.tics.2017.09.005.

Botvinick, Matthew et al. (Nov. 1999). "Conflict monitoring versus selection for-action in anterior cingulate cortex". In: *Nature* 402.6758, pp. 179–181. ISSN: 00280836. DOI: 10.1038/46035. URL: www.nature.com.

Brannon, Elizabeth M (2006). "The representation of numerical magnitude". In: *Current Opinion in Neurobiology* 16.2, pp. 222–229. ISSN: 09594388. DOI: 10.1016/j.conb.2006.03.002. URL: www.sciencedirect.com.

Camilleri, J A et al. (2018). "Definition and characterization of an extended multiple-demand network". In: *NeuroImage* 165, pp. 138–147. ISSN: 10959572. DOI: 10.1016/j.neuroimage.2017.10.020. URL: https://doi.org/10.1016/j.neuroimage.2017.10.020.

Carruthers, Sarah, Michael E J Masson, and Ulrike Stege (2012). "Human Performance on Hard Non-Euclidean Graph Problems: Vertex Cover". In: *The Journal of Problem Solving* 5.1, p. 34. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1142.

Cheeseman, Peter, Bob Kanefsky, and William M Taylor (1991). "Where the Really Hard Problems Are". In: *The 12nd International Joint Conference on Artificial Intelligence*, pp. 331–337. ISBN: 1-55860-160-0. DOI: 10.1.1.97.3555.

Chernev, Alexander, Ulf Böckenholt, and Joseph Goodman (2015). "Choice overload: A conceptual review and meta-analysis". In: *Journal of Consumer Psychology* 25, pp. 333–358. DOI: 10.1016/j.jcps.2014.08.002.

Chu, Yun and James N. MacGregor (2011). "Human Performance on Insight Problem Solving: A Review". In: *The Journal of Problem Solving* 3.2, p. 119. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1094. URL: http://dx.doi.org/10.7771/1932-6246.1094.

Church, Alonzo (1936). "An unsolvable problem of elementary number theory". In: *American Journal of Mathematics* 58.2, pp. 345–363. ISSN: 00029327. DOI: 10.2307/2371045. URL: http://www.jstor.org/stable/2371045.

Cicirello, Vincent A. (Mar. 2019). "Classification of permutation distance metrics for fitness landscape analysis". In: *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*. Vol. 289. Springer Verlag, pp. 81–97. ISBN: 9783030242015. DOI: 10.1007/978-3-030-24202-2{_}7.

Cook, Stephen A. (1971). "The complexity of theorem-proving procedures". In: *Proceedings of the third annual ACM symposium on Theory of computing - STOC '71*, pp. 151–158. DOI: 10.1145/800157.805047.

Coste, Clio P and Andreas Kleinschmidt (2016). "Cingulo-opercular network activity maintains alertness". In: *NeuroImage* 128, pp. 264–272. ISSN: 10959572. DOI: 10.1016/j.neuroimage.2016.01.026. URL: http://dx.doi.org/10.1016/j.neuroimage.2016.01.026.

Crittenden, Ben M and John Duncan (2014). "Task difficulty manipulation reveals multiple demand activity but no frontal lobe hierarchy". In: *Cerebral Cortex* 24.2, pp. 532–540. ISSN: 14602199. DOI: 10.1093/cercor/bhs333. URL: https://academic.oup.com/cercor/article/24/2/532/354842.

Crittenden, Ben M, Daniel J Mitchell, and John Duncan (2016). "Task encoding across the multiple demand cortex is consistent with a frontoparietal and cingulo-opercular dual networks distinction". In: *Journal of Neuroscience* 36.23, pp. 6147–6155. ISSN: 15292401. DOI: 10.1523/JNEUROSCI.4590-15.2016.

Crottaz-Herbette, S and V Menon (2006). "Where and when the anterior cingulate cortex modulates attentional response: Combined fMRI and ERP evidence". In: *Journal of Cognitive Neuroscience* 18.5, pp. 766–780. ISSN: 0898929X. DOI: 10.1162/jocn.2006.18.5.766. URL: http://psyscope.psy.cmu.edu.

De Smedt, Bert, Ian D Holloway, and Daniel Ansari (2011). "Effects of problem size and arithmetic operation on brain activation during calculation in children with varying levels of arithmetical fluency". In: *NeuroImage* 57.3, pp. 771–781. ISSN: 10538119. DOI: 10.1016/j.neuroimage.2010.12.037.

De Visscher, Alice and Marie Pascale Noël (2014). "The detrimental effect of interference in multiplication facts storing: Typical development and individual differences". In: *Journal of Experimental Psychology: General* 143.6, pp. 2380–2400. ISSN: 00963445. DOI: 10.1037/xge0000029.

Demaine, Erik, Joseph Mitchell, and Joseph O'Rouke (2004). *The Open Problems Project - Problem 54: Traveling Salesman Problem in Solid Grid Graphs*. URL: https://topp.openproblem.net/.

Dosenbach, Nico U.F., Damien A Fair, Alexander L Cohen, et al. (2008). "A dual-networks architecture of top-down control". In: *Trends in Cognitive Sciences* 12.3, pp. 99–105. ISSN: 13646613. DOI: 10.1016/j.tics.2008.01.001.

Dosenbach, Nico U.F., Damien A Fair, Francis M Miezin, et al. (2007). "Distinct brain networks for adaptive and stable task control in humans". In: *Proceedings*

of the National Academy of Sciences of the United States of America 104.26, pp. 11073–11078. ISSN: 00278424. DOI: 10.1073/pnas.0704320104.

Dosenbach, Nico U.F., Kristina M. Visscher, et al. (June 2006). "A Core System for the Implementation of Task Sets". In: *Neuron* 50.5, pp. 799–812. ISSN: 08966273. DOI: 10.1016/j.neuron.2006.04.031.

Dry, Matthew, Michael D Lee, et al. (2006). "Human Performance on Visually Presented Traveling Salesperson Problems with Varying Numbers of Nodes". In: *The Journal of Problem Solving* 1.1. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1004. URL: http://dx.doi.org/10.7771/1932-6246.1004.

Dry, Matthew, Kym Preiss, and Johan Wagemans (Feb. 2012). "Clustering, Randomness, and Regularity: Spatial Distributions and Human Performance on the Traveling Salesperson Problem and Minimum Spanning Tree Problem". In: *The Journal of Problem Solving* 4.1, p. 1. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1117. URL: http://dx.doi.org/10.7771/1932-6246.1117.

Dubis, Joseph W et al. (2016). "Tasks Driven by Perceptual Information Do Not Recruit Sustained BOLD Activity in Cingulo-Opercular Regions". In: *Cerebral Cortex* 26.1, pp. 192–201. ISSN: 14602199. DOI: 10.1093/cercor/bhu187.

Duncan, John (2010). "The multiple-demand (MD) system of the primate brain: mental programs for intelligent behaviour". In: *Trends in Cognitive Sciences* 14.4, pp. 172–179. ISSN: 13646613. DOI: 10.1016/j.tics.2010.01.004. URL: http://dx.doi.org/10.1016/j.tics.2010.01.004.

Duncan, John and Adrian M Owen (2000). *Common regions of the human frontal lobe recruited by diverse cognitive demands*. DOI: 10.1016/S0166-2236(00)01633-7.

Duverne, Sandrine and Etienne Koechlin (2017). "Rewards and Cognitive Control in the Human Prefrontal Cortex". In: *Cerebral Cortex* 27.10, pp. 5024–5039. ISSN: 14602199. DOI: 10.1093/cercor/bhx210. URL: https://academic.oup.com/cercor/article/27/10/5024/4080829.

Fedorenko, Evelina, John Duncan, and Nancy Kanwisher (2013). "Broad domain generality in focal regions of frontal and parietal cortex". In: *Proceedings of the National Academy of Sciences of the United States of America* 110.41, pp. 16616–16621. ISSN: 00278424. DOI: 10.1073/pnas.1315235110.

Fehr, Ernst and Klaus M. Schmidt (Aug. 1999). "A theory of fairness, competition, and cooperation". In: *Quarterly Journal of Economics* 114.3, pp. 817–868. ISSN: 00335533. DOI: 10.1162/003355399556151. URL: https://academic.oup.com/qje/article-lookup/doi/10.1162/003355399556151.

Fimbel, Eric, Stéphane Lauzon, and Constant Rainville (Sept. 2009). "Performance of humans vs. exploration algorithms on the Tower of London Test". In: *PLoS ONE* 4.9. Ed. by Josh Bongard, e7263. ISSN: 19326203. DOI: 10.1371/journal.pone.0007263. URL: https://dx.plos.org/10.1371/journal.pone.0007263.

Frixione, Marcello (2001). "Tractable competence". In: *Minds and Machines* 11.3, pp. 379–397. ISSN: 09246495. DOI: 10.1023/A:1017503201702.

Garain, D.N and Sanjeev Kumar (2018). "Japanese vs Vedic Methods for Multiplication". In: *International Journal of Mathematics Trends and Technology* 54.3, pp. 228–235. ISSN: 2231-5373. DOI: 10.14445/22315373/ijmtt-v54p525. URL: http://www.ijmttjournal.org.

Gent, Ian P and Toby Walsh (1996). "The TSP phase transition". In: *Artificial Intelligence* 88.1-2, pp. 349–358. ISSN: 00043702. DOI: 10.1016/S0004-3702(96)00030-6.

Gent, Ian P. et al. (1996). "The constrainedness of search". In: *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 1*. Portland. Oregon, pp. 246–252.

Gershman, Samuel J, Eric J Horvitz, and Joshua B Tenenbaum (2015). "Computational rationality: A converging paradigm for intelligence in brains, minds, and machines". In: *Science* 349.6245, pp. 273–278. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.aac6076.

Gigerenzer, Gerd and Wolfgang Gaissmaier (Jan. 2011). "Heuristic decision making". In: *Annual Review of Psychology* 62, pp. 451–482. ISSN: 00664308. DOI: 10.1146/annurev-psych-120709-145346.

Gigerenzer, Gerd. and Reinhard. Selten (2001). *Bounded rationality : the adaptive toolbox*. MIT Press, p. 377. ISBN: 9780262072144.

Grabner, Roland H et al. (2009). "To retrieve or to calculate? Left angular gyrus mediates the retrieval of arithmetic facts during problem solving". In: *Neuropsychologia* 47.2, pp. 604–608. ISSN: 00283932. DOI: 10.1016/j.neuropsychologia.2008.10.013.

Gratton, Caterina, Haoxin Sun, and Steven E. Petersen (Mar. 2018). "Control networks and hubs". In: *Psychophysiology* 55.3, e13032. ISSN: 14698986. DOI: 10.1111/psyp.13032. URL: http://doi.wiley.com/10.1111/psyp.13032.

Gratton, Gabriele et al. (Mar. 2018). "Dynamics of cognitive control: Theoretical bases, paradigms, and a view for the future". In: *Psychophysiology* 55.3. ISSN: 14698986. DOI: 10.1111/psyp.13016.

Griffiths, Thomas L., Falk Lieder, and Noah D. Goodman (2015). "Rational use of cognitive resources: Levels of analysis between the computational and the algorithmic". In: *Topics in Cognitive Science* 7.2, pp. 217–229. ISSN: 17568765. DOI: 10.1111/tops.12142.

Guid, Matej and Ivan Bratko (2013). "Search-Based Estimation of Problem Difficulty for Humans". In: *Artificial Intelligence in Education. AIED 2013. Lecture Notes in Computer Science*. Ed. by Lane H.C. et al. Vol. 7926. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-642-39112-5{_}131.

Hill, Arthur V. (Aug. 1982). "An experimental comparison of human schedulers and heuristic algorithms for the traveling salesman problem". In: *Journal of Operations Management* 2.4, pp. 215–223. ISSN: 0272-6963. DOI: 10.1016/0272-6963(82)90010-9.

Hirtle, Stephen C. and Tommy Gärling (May 1992). "Heuristic rules for sequential spatial decisions". In: *Geoforum* 23.2, pp. 227–238. ISSN: 00167185. DOI: 10.1016/0016-7185(92)90019-Z.

Holroyd, Clay B. and Nick Yeung (Feb. 2012). "Motivation of extended behaviors by anterior cingulate cortex". In: *Trends in Cognitive Sciences* 16.2, pp. 122–128. ISSN: 13646613. DOI: 10.1016/j.tics.2011.12.008.

Kahneman, Daniel and Amos Tversky (1979). "Prospect Theory: An Analysis of Decision under Risk". In: *Econometrica* 47.2, pp. 263–292. ISSN: 00129682. DOI: 10.2307/1914185.

Kellerer, Hans, Ulrich Pferschy, and David Pisinger (2004). *Knapsack Problems*. Berlin, Heidelberg: Springer Berlin Heidelberg, p. 546. ISBN: 978-3-642-07311-3. DOI: 10.1007/978-3-540-24777-7.

Knoblich, Günther et al. (1999). "Constraint Relaxation and Chunk Decomposition in Insight Problem Solving". In: *Journal of Experimental Psychology: Learning Memory and Cognition* 25.6, pp. 1534–1555. ISSN: 02787393. DOI: 10.1037/0278-7393.25.6.1534.

Koechlin, Etienne (2016). "Prefrontal executive function and adaptive behavior in complex environments". In: *Current Opinion in Neurobiology* 37, pp. 1–6. ISSN: 18736882. DOI: 10.1016/j.conb.2015.11.004. URL: http://dx.doi.org/10.1016/j.conb.2015.11.004.

Krzakala, Florent, Andrea Montanari, et al. (June 2006). "Gibbs States and the Set of Solutions of Random Constraint Satisfaction Problems". In: *Proceedings of the National Academy of Sciences of the United States of America* 104.25, pp. 10318–23. ISSN: 0027-8424. DOI: 10.1073/pnas.0703685104.

Krzakala, Florent and Lenka Zdeborová (Nov. 2007). "Phase Transitions and Computational Difficulty in Random Constraint Satisfaction Problems". In: *Proceedings of the International Workshop on Statistical-Mechanical Informatics*. DOI: 10.1088/1742-6596/95/1/012012.

Kwisthout, Johan, Todd Wareham, and Iris Van Rooij (July 2011). "Bayesian Intractability Is Not an Ailment That Approximation Can Cure". In: *Cognitive Science* 35.5, pp. 779–784. ISSN: 03640213. DOI: 10.1111/j.1551-6709.2011.01182.x.

Levin, Leonid A (1973). "Universal Search Problems". In: *Problemy Peredachi Informatsii* 9.3, pp. 115–116. ISSN: 1058-6180.

Lewis, Richard L., Andrew Howes, and Satinder Singh (2014). "Computational rationality: Linking mechanism and behavior through bounded utility maximization". In: *Topics in Cognitive Science* 6.2, pp. 279–311. ISSN: 17568765. DOI: 10.1111/tops.12086.

Lieder, Falk and Thomas L. Griffiths (Nov. 2017). "Strategy selection as rational metareasoning." In: *Psychological Review* 124.6, pp. 762–794. ISSN: 1939-1471. DOI: 10.1037/rev0000075.

MacGregor, James N. and Yun Chu (2011). "Human Performance on the Traveling Salesman and Related Problems: A Review". In: *The Journal of Problem Solving* 3.2, p. 1. ISSN: 1932-6246. DOI: 10.7771/1932-6246.1090. URL: http://dx.doi.org/10.7771/1932-6246.1090.

MacGregor, James N., Thomas C. Ormerod, and Edward P. Chronicle (1999). "Spatial and contextual factors in human performance on the travelling salesperson problem". In: *Perception* 28.11, pp. 1417–1427. ISSN: 03010066. DOI: 10.1068/p2863.

Marek, Scott and Nico U.F. Dosenbach (Jan. 2019). "Control networks of the frontal lobes". In: *Handbook of Clinical Neurology*. Vol. 163. Elsevier B.V., pp. 333–347. DOI: 10.1016/B978-0-12-804281-6.00018-5.

Matejko, Anna A. and Daniel Ansari (Dec. 2018). "Contributions of functional Magnetic Resonance Imaging (fMRI) to the study of numerical cognition". In: *Journal of Numerical Cognition* 4.3, pp. 505–525. ISSN: 2363-8761. DOI: 10.5964/jnc.v4i3.136. URL: https://jnc.psychopen.eu/article/view/136.

McClamrock, Ron (May 1991). "Marr's three levels: A re-evaluation". In: *Minds and Machines* 1.2, pp. 185–196. ISSN: 09246495. DOI: 10.1007/BF00361036. URL: https://link.springer.com/article/10.1007/BF00361036.

Miller, Earl K. and Jonathan D. Cohen (Mar. 2001). "An integrative theory of prefrontal cortex function". In: *Annual Review of Neuroscience* 24.1, pp. 167–202. ISSN: 0147006X. DOI: 10.1146/annurev.neuro.24.1.167. URL: http://www.annualreviews.org/doi/10.1146/annurev.neuro.24.1.167.

Mitchell, David, Bart Selman, and Hector Levesque (1992). "Hard and Easy Distributions of SAT Problems". In: *Proceedings of the National Conference on Artificial Inteligence (AAAI)*. April, pp. 459–465. ISBN: 0-262-51063-4. DOI: 10.1.1.52.6632.

Miyake, Akira et al. (Aug. 2000). "The Unity and Diversity of Executive Functions and Their Contributions to Complex "Frontal Lobe" Tasks: A Latent Variable Analysis". In: *Cognitive Psychology* 41.1, pp. 49–100. ISSN: 00100285. DOI: 10.1006/cogp.1999.0734.

Monasson, Remi et al. (1999). "Determining computational complexity from characteristic 'phase transitions'". In: *Nature* 400.6740, pp. 133–137. ISSN: 0028-0836. DOI: 10.1038/22055.

Monasson, Rémi and Riccardo Zecchina (1996). "Entropy of the K-satisfiability problem". In: *Physical Review Letters* 76.21, pp. 3881–3885. ISSN: 10797114. DOI: 10.1103/PhysRevLett.76.3881.

Moore, Cristopher and Stephan Mertens (2011). *The Nature of Computation*. 1st ed. Oxford University Press, p. 1004. ISBN: 9780199233212. DOI: 10.1093/acprof:oso/9780199233212.001.0001.

Moser, I., M. Gheorghita, and A. Aleti (Sept. 2017). "Identifying features of fitness landscapes and relating them to problem difficulty". In: *Evolutionary Computation* 25.3, pp. 407–437. ISSN: 15309304. DOI: 10.1162/EVCO{_}a{_}00177. URL: https://www.mitpressjournals.org/doix/abs/10.1162/evco_a_00177.

Murawski, Carsten and Peter Bossaerts (2016). "How Humans Solve Complex Problems: The Case of the Knapsack Problem". In: *Nature (Scientific Reports)* 6.34851. ISSN: 2045-2322. DOI: 10.1038/srep34851.

Nash, John F (1950). "Equilibrium points in n-person games". In: *Proceedings of the National Academy of Sciences of the United States of America* 36.1, pp. 48–49. ISSN: 0027-8424. DOI: 10.1073/pnas.36.1.48. URL: http://www.pnas.org/cgi/doi/10.1073/pnas.36.1.48.

Neta, Maital, Steven M Nelson, and Steven E Petersen (2017). "Dorsal Anterior Cingulate, Medial Superior Frontal Cortex, and Anterior Insula Show Performance Reporting-Related Late Task Control Signals". In: *Cerebral cortex* 27.3, pp. 2154–2165. ISSN: 14602199. DOI: 10.1093/cercor/bhw053.

Neta, Maital, Bradley L Schlaggar, and Steven E Petersen (2014). "Separable responses to error, ambiguity, and reaction time in cingulo-opercular task control regions". In: *NeuroImage* 99, pp. 59–68. ISSN: 10959572. DOI: 10.1016/j.neuroimage.2014.05.053. URL: http://dx.doi.org/10.1016/j.neuroimage.2014.05.053.

Newell, Ben R., Nicola J. Weston, and David R. Shanks (May 2003). "Empirical tests of a fast-and-frugal heuristic: Not everyone "takes-the-best"". In: *Organizational*

Behavior and Human Decision Processes 91.1, pp. 82–96. ISSN: 07495978. DOI: 10.1016/S0749-5978(02)00525-3.

Nitschke, Kai et al. (Jan. 2017). "A Meta-analysis on the neural basis of planning: Activation likelihood estimation of functional brain imaging results in the Tower of London task". In: *Human Brain Mapping* 38.1, pp. 396–413. ISSN: 10970193. DOI: 10.1002/hbm.23368.

Nomura, Emi M et al. (2010). "Double dissociation of two cognitive control networks in patients with focal brain lesions". In: *Proceedings of the National Academy of Sciences of the United States of America* 107.26, pp. 12017–12022. ISSN: 00278424. DOI: 10.1073/pnas.1002431107. URL: www.pnas.org/cgi/doi/10.1073/pnas.1002431107.

Nudelman, Eugene et al. (2004). "Understanding random SAT: Beyond the clauses-to-variables ratio". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 3258, pp. 438–452. ISSN: 16113349. DOI: 10.1007/978-3-540-30201-8{_}33. URL: https://link.springer.com/chapter/10.1007/978-3-540-30201-8_33.

Ohlsson, Stellan (2012). "The Problems with Problem Solving: Reflections on the Rise, Current Status, and Possible Future of a Cognitive Research Paradigm 1". In: *The Journal of Problem Solving* 5.1. DOI: 10.7771/1932-6246.1144. URL: http://dx.doi.org/10.7771/1932-6246.1144.

Otworowska, Maria et al. (2018). "Demons of Ecological Rationality". In: *Cognitive Science* 42.3, pp. 1057–1066. ISSN: 0364-0213. DOI: 10.1111/cogs.12530.

Payne, John W., James R. Bettman, and Eric J. Johnson (1993). *The Adaptive Decision Maker*. DOI: 10.1017/cbo9781139173933.

Posner, Michael I and Steven E Petersen (1990). *The attention system of the human brain*. DOI: 10.1146/annurev.ne.13.030190.000325. URL: www.annualreviews.org.

Power, Jonathan D and Steven E Petersen (2013). *Control-related systems in the human brain*. DOI: 10.1016/j.conb.2012.12.009. URL: http://dx.doi.org/10.1016/j.conb.2012.12.009.

Power, Jonathan D. et al. (Nov. 2011). "Functional Network Organization of the Human Brain". In: *Neuron* 72.4, pp. 665–678. ISSN: 08966273. DOI: 10.1016/j.neuron.2011.09.006.

Pudlák, Pavel (2013). *Logical foundations of mathematics and computational complexity: a gentle introduction*. 1st ed. Springer International Publishing, p. 695. ISBN: 9783319001180. DOI: 10.1007/978-3-319-00119-7{_}1.

Rescorla, Michael (2020). *The Computational Theory of Mind*. Ed. by Edward N. Zalta. URL: https://plato.stanford.edu/archives/fall2020/entries/computational-mind/.

Rich, Patricia et al. (2019). "Naturalism, tractability and the adaptive toolbox". In: *Synthese*, pp. 1–36. DOI: 10.1007/s11229-019-02431-2.

Ruocco, Anthony C et al. (2014). "A problem-solving task specialized for functional neuroimaging: Validation of the Scarborough adaptation of the Tower of London (S-TOL) using near-infrared spectroscopy". In: *Frontiers in Human Neuroscience* 8.MAR. ISSN: 16625161. DOI: 10.3389/fnhum.2014.00185. URL: www.frontiersin.org.

Sadaghiani, Sepideh and Mark D'Esposito (2015). "Functional characterization of the cingulo-opercular network in the maintenance of tonic alertness". In: *Cerebral Cortex* 25.9, pp. 2763–2773. ISSN: 14602199. DOI: 10.1093/cercor/bhu072. URL: https://academic.oup.com/cercor/article/25/9/2763/2926085.

Sahni, Sartaj and Sartaj (Jan. 1975). "Approximate Algorithms for the 0/1 Knapsack Problem". In: *Journal of the ACM* 22.1, pp. 115–124. ISSN: 00045411. DOI: 10.1145/321864.321873.

Samuelson, P. A. (1938). "A Note on the Pure Theory of Consumer's Behaviour". In: *Economica* 5.17, pp. 61–71. ISSN: 00130427. DOI: 10.2307/2548836.

Schiavinotto, Tommaso and Thomas Stützle (Oct. 2007). "A review of metrics on permutations for search landscape analysis". In: *Computers and Operations Research* 34.10, pp. 3143–3153. ISSN: 03050548. DOI: 10.1016/j.cor.2005.11.022.

Seeley, William W. et al. (Feb. 2007). "Dissociable intrinsic connectivity networks for salience processing and executive control". In: *Journal of Neuroscience* 27.9, pp. 2349–2356. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.5587-06.2007.

Selman, Bart and Scott Kirkpatrick (Mar. 1996). "Critical behavior in the computational cost of satisfiability testing". In: *Artificial Intelligence* 81.1-2, pp. 273–295. ISSN: 0004-3702. DOI: 10.1016/0004-3702(95)00056-9.

Semerjian, Guilhem (2008). "On the freezing of variables in random constraint satisfaction problems". In: *Journal of Statistical Physics* 130.2, pp. 251–293. ISSN: 00224715. DOI: 10.1007/s10955-007-9417-7.

Sestieri, Carlo et al. (2014). "Domain-general signals in the cingulo-opercular network for visuospatial attention and episodic memory". In: *Journal of Cognitive Neuroscience* 26.3, pp. 551–568. ISSN: 15308898. DOI: 10.1162/jocn{_}a{_}00504.

Shallice, T (1982). "Specific impairments of planning." In: *Philosophical transactions of the Royal Society of London. Series B, Biological sciences* 298.1089, pp. 199–209. ISSN: 09628436. DOI: 10.1098/rstb.1982.0082. URL: https://www.jstor.org/stable/2395870.

Shefrin, Hersh and Meir Statman (1985). "The Disposition to Sell Winners Too Early and Ride Losers Too Long: Theory and Evidence". In: *The Journal of Finance* 40.3, pp. 777–790. ISSN: 15406261. DOI: 10.1111/j.1540-6261.1985.tb05002.x.

Shenhav, Amitai, Matthew M. Botvinick, and Jonathan D. Cohen (2013). "The expected value of control: An integrative theory of anterior cingulate cortex function". In: *Neuron* 79.2, pp. 217–240. ISSN: 08966273. DOI: 10.1016/j.neuron.2013.07.007. URL: http://dx.doi.org/10.1016/j.neuron.2013.07.007.

Shepard, Roger N. and Jacqueline Metzler (Feb. 1971). "Mental rotation of three-dimensional objects". In: *Science* 171.3972, pp. 701–703. ISSN: 00368075. DOI: 10.1126/science.171.3972.701.

Silvetti, Massimo et al. (Aug. 2018). "Dorsal anterior cingulate-brainstem ensemble as a reinforcement meta-learner". In: *PLoS Computational Biology* 14.8, e1006370. ISSN: 15537358. DOI: 10.1371/journal.pcbi.1006370.

Simon, Herbert A (1956). "Rational choice and the structure of the environment". In: *Psychological Review* 63.2, pp. 129–138. ISSN: 0033295X. DOI: 10.1037/h0042769.

Simon, Herbert A (1990). "Invariants of human behavior". In: *Annual Review of Psychology* 41.1, pp. 1–19. ISSN: 00664308. DOI: 10.1146/annurev.psych.41.1.1. URL: www.annualreviews.org.

Sprugnoli, Giulia et al. (May 2017). *Neural correlates of Eureka moment.* DOI: 10.1016/j.intell.2017.03.004.

Stazyk, Edmund H., Mark H. Ashcraft, and Mary S. Hamann (1982). "A network approach to mental multiplication". In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* 8.4, pp. 320–335. ISSN: 02787393. DOI: 10.1037/0278-7393.8.4.320.

Tavares, Jorge, Francisco B. Pereira, and Ernesto Costa (June 2008). "Multidimensional knapsack problem: A fitness landscape analysis". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 38.3, pp. 604–616. ISSN: 10834419. DOI: 10.1109/TSMCB.2008.915539.

Todd, Peter M and Gerd Gigerenzer (2012). *Ecological rationality: Intelligence in the world.* Evolution and cognition. Todd, Peter M.: Cognitive Science Program, Indiana University, 1101 E. 10th St., Bloomington, IN, US, 47405, peter.m.todd@gmail.com: Oxford University Press, pp. xviii, 590–xviii, 590. ISBN: 978-0-19-531544-8 (Hardcover). DOI: 10.1093/acprof:oso/9780195315448.001.0001.

Tsotsos, John K (1988). "How does human vision beat the computational complexity of visual perception". In: *Computational processes in human vision: an interdisciplinary perspective.* Ed. by Z.W. Pylyshyn (Ed.) Ablex Pub. Corp, pp. 286–338.

Tsotsos, John K. (1990). "Analyzing vision at the complexity level". In: *Behavioral and Brain Sciences* 13.3, pp. 423–445. ISSN: 14691825. DOI: 10.1017/S0140525X00079577.

Turing, A M (1937). "On Computable Numbers, With Application to the Entscheidungs Problem". In: *Proceedings of the London Mathematical Society*, pp. 230–265. DOI: 10.1112/plms/s2-42.1.23.

Tversky, Amos and Daniel Kahneman (1981). "The framing of decisions and the psychology of choice". In: *Science* 211.4481, pp. 453–458. ISSN: 00368075. DOI: 10.1126/science.7455683.

Tversky, Amos and Daniel Kahneman (Oct. 1992). "Advances in prospect theory: Cumulative representation of uncertainty". In: *Journal of Risk and Uncertainty* 5.4, pp. 297–323. ISSN: 0895-5646. DOI: 10.1007/BF00122574. URL: http://link.springer.com/10.1007/BF00122574.

Van Opheusden, Bas and Wei Ji Ma (2019). *Tasks for aligning human and machine planning.* DOI: 10.1016/j.cobeha.2019.07.002. URL: https://doi.org/10.1016/j.cobeha.2019.07.002.

Van Rooij, Iris (2008). "The Tractable Cognition Thesis". In: *Cognitive Science: A Multidisciplinary Journal* 32.6, pp. 939–984. ISSN: 0364-0213. DOI: 10.1080/03640210801897856. URL: http://doi.wiley.com/10.1080/03640210801897856.

Van Rooij, Iris, Mark Blokpoel, et al. (Apr. 2019). *Cognition and Intractability.* Cambridge University Press. DOI: 10.1017/9781107358331.

Van Rooij, Iris, Ulrike Stege, and Helena Kadlec (2005). "Sources of complexity in subset choice". In: *Journal of Mathematical Psychology* 49.2, pp. 160–187. ISSN: 00222496. DOI: 10.1016/j.jmp.2005.01.002. URL: www.elsevier.com/locate/jmp.

Van Rooij, Iris, Cory D. Wright, et al. (Feb. 2018). "Rational analysis, intractability, and the prospects of 'as if'-explanations". In: *Synthese* 195.2, pp. 491–510. ISSN: 0039-7857. DOI: 10.1007/s11229-014-0532-0. URL: http://link.springer.com/10.1007/s11229-014-0532-0.

Vassena, Eliana, Clay B. Holroyd, and William H. Alexander (2017). "Computational models of anterior cingulate cortex: At the crossroads between prediction and effort". In: *Frontiers in Neuroscience* 11.JUN, pp. 1–9. ISSN: 1662453X. DOI: 10.3389/fnins.2017.00316.

Verguts, Tom, Eliana Vassena, and Massimo Silvetti (Mar. 2015). "Adaptive effort investment in cognitive and physical tasks: A neurocomputational model". In: *Frontiers in Behavioral Neuroscience* 9, p. 57. ISSN: 16625153. DOI: 10.3389/fnbeh.2015.00057.

Westbrook, Andrew and Todd S Braver (2015). "Cognitive effort: A neuroeconomic approach." In: *Cognitive, affective & behavioral neuroscience* 15.2, pp. 395–415. ISSN: 1531-135X.

Yadav, Nitin et al. (2020). "Is Hardness Inherent In Computational Problems? Performance Of Human And Digital Computers On Random Instances Of The 0-1 Knapsack Problem". In: *24th European Conference on Artificial Intelligence (ECAI 2020).*

Yoo, Seng Bum Michael, Benjamin Yost Hayden, and John M Pearson (2021). *Continuous decisions*. DOI: 10.1098/rstb.2019.0664. URL: https://doi.org/10.1098/rstb.2019.0664.

Zdeborová, Lenka and Florent Krzakala (2007). "Phase transitions in the coloring of random graphs". In: *Physical Review E* 76.3. ISSN: 15393755. DOI: 10.1103/PhysRevE.76.031131.

Zdeborová, Lenka and Marc Mézard (Dec. 2008). "Constraint satisfaction problems with isolated solutions are hard". In: *Journal of Statistical Mechanics: Theory and Experiment* 2008.12, P12004. ISSN: 1742-5468. DOI: 10.1088/1742-5468/2008/12/P12004.

Zweig, Katharina A., Gergely Palla, and Tamás Vicsek (Apr. 2010). "What makes a phase transition? Analysis of the random satisfiability problem". In: *Physica A: Statistical Mechanics and its Applications* 389.8, pp. 1501–1511. DOI: 10.1016/J.PHYSA.2009.12.051.