20

Search Environments, Representation, and Encoding

Lael J. Schooler, Curt Burgess, Robert L. Goldstone,
Wai-Tat Fu, Sergey Gavrilets, David Lazer,
James A. R. Marshall, Frank Neumann, and Jan M. Wiener

Abstract

This chapter explores the benefits of restructuring search spaces and internal representations so as to make search more efficient. It begins by providing a formal definition of search, and proposes a method for shifting search between low- and high-dimensionality problem spaces. Consideration is given to how learning shapes the representations that help people search efficiently as well as on constraints that people face. Some constraints are considered biases necessary to make sense out of the world; others (e.g., working memory) are taken as both "limiters" to be overcome and "permitters" that make learning in a finite amount of time possible at all. Further constraints on search are tied to the physical structure of the world. The chapter concludes with a discussion of social search, where communication can promote exploration and exploitation in an environment that often consists of other agents searching for similar solutions.

Introduction

In 1975, Allen Newell and Herbert Simon received the Turing award for their contributions to computer science and psychology. In large part, they were being honored for their work in artificial intelligence. In their acceptance address, Newell and Simon (1976, 1987) described how they approached problems in artificial intelligence by studying the natural intelligence of people. These studies of humans and machines led them to conclude that the key to intelligence was the ability to manipulate symbols. They believed that all intelligent behavior, whether human or machine, arises from composing symbols into entities called symbol structures that can be manipulated by prescribed sets of operators. Some operators can construct new structures, whereas others modify or destroy existing ones. The combination of symbol structures and the corresponding operators define what Newell and Simon called a symbol

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

system. These symbol systems were at the heart of their *physical symbol system hypothesis*: "A physical symbol system has the necessary and sufficient means for general intelligent action" (Newell and Simon 1987:293).

Their claim with the physical symbol system hypothesis (or symbol system hypothesis; Newell 1980) was that symbol systems not only support intelligent behavior, they are essential for the display of intelligent behavior. They viewed the symbol system hypothesis as the guiding principle that should organize research on human and artificial intelligence. For Newell and Simon, symbol systems were as fundamental to the study of intelligence as the theory of plate tectonics is to geology or germ theory is to the study of disease. This is a radical view, but one that was proposed as a hypothesis to be tested. They were, after all, empiricists at heart. The hypothesis that intelligent behavior rests on symbol structures flowing one into the next, transformed by operators, led them to see problem solving as search through a space of symbol structures that represent possible solutions to particular problems. Thus, intelligent behavior was a form of search in a problem space of symbol structures.

Their belief in the fundamental importance of search led to their second guiding principle, the *heuristic search hypothesis*: "A physical symbol system exercises its intelligence in problem solving by search; that is, by generating and progressively modifying symbol structures until it produces a solution structure" (Newell and Simon 1987:230). Their *general problem solver* (GPS) algorithm worked by transforming one solution into the next until a dead end (requiring back-tracking) or goal was reached. Just as a rat might search for food in a field by moving from patch to patch, GPS moved in an abstract solution space from symbol set to symbol set. They argued that problem solving must depend on "heuristic (i.e., knowledge-controlled) search" (Newell 1980), because intelligent behavior can be observed even when problem spaces are so vast that they cannot be exhaustively searched. The importance of basic notions of symbol systems and heuristic search in our report is a testament to the lasting legacy of Newell and Simon's formalization of search processes.

We start by providing a formal definition of search. Inspired by results showing that high-dimensionality spaces imply that good solutions should be well connected to each other, we propose a method for shifting search between low- and high-dimensionality problem spaces. Turning from formal methods to people, we consider the ways in which learning shapes the representations that help people search efficiently. Thereafter we discuss constraints that people face: some are considered biases necessary to make sense out of the world; others (e.g., working memory) are taken as both "limiters" to be overcome and "permitters" that make learning in a finite amount of time possible at all. Further constraints on search are tied to the physical structure of the world. Finally, we turn to social search, which complements heuristic search by supplementing internal cognitive constraints on search within an individual with the constraints provided by an environment that often consists of other agents searching for similar solutions.

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

A Formal Definition of Search

To aid concrete discussion of iterative search algorithms, we define search problems in a formal way that is consistent with Newell and Simon's notion of symbol systems. A search problem is given by a triplet (S, f, W), where S is the considered search space, f: $S \rightarrow R$ is a function assigning objective values to the elements of S (representing all possible solutions), and W is a set of constraints.

To illustrate this, let us consider the well-known traveling salesman problem (TSP). Input is given by a set of n cities $\{1, \ldots, n\}$, and between each pair of cities, i and j, there is a distance, $d_{i,j}$. A tour in the TSP problem visits each city exactly once and returns to the origin. We focus on two variants:

1. Satisficing version: Is there a tour of cost at most k?
2. Optimization version: Find a tour of minimal cost.

To fit the TSP problem into our search framework, the search space S is given by all permutations of the n cities (i.e., ordered tours through all the cities, as opposed to the locations of the individual cities themselves in physical space). The cost of a permutation π is then computed by starting at the first city in the permutation, $\pi(1)$, moving to the second city in the permutation, $\pi(2)$, then to the third, $\pi(3)$, and so on. The cost of this permutation is given by the sum of the distances traveled to construct the tour:

$$\text{cost}(\pi) = d_{\pi(1),\pi(2)} + d_{\pi(2),\pi(3)} + \ldots + d_{\pi(n-1),\pi(n)} + d_{\pi(n),\pi(1)}. \quad (20.1)$$

Let us now consider optimization problems tackled by iterative search algorithms. The task is to find an element x^* in S which minimizes the function value:

$$x^* = \arg\min_{x \in S} f(x). \quad (20.2)$$

In the TSP example, we would search through the space S for a tour that has minimal cost. To apply iterative search algorithms to optimization problems, three steps are necessary:

1. Choose a representation of the elements in the search space S.
2. Define a fitness function (might be different from f) that assigns fitness values to points in the search space S.
3. Define operators that construct, from a set of solutions, a new set of solutions. The combination of the search representation in step (1) and the operators gives a structure to the search space in terms of how local neighbors in the search space are related to each other.

This framework fits many successful algorithms for optimization, such as local search and simulated annealing. Furthermore, many successful bio-inspired algorithms (e.g., evolutionary algorithms, ant colony optimization, and particle

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

swarm optimization) fit into this framework. They differ from each other in the representation chosen and the operators used to produce new solutions. As we have seen, possible solutions for the TSP problem can be represented by permutations of the *n* cities. Furthermore, the fitness assignment can be straightforward by taking the length of the tour that is encoded by the permutation. Given a permutation of the input elements, we next have to think about what operators could be used to construct a new solution.

A well-known operator for solving the TSP problem is the state-of-the-art 2-OPT operator. It takes the current tour, chooses two edges of the tour (i.e., connections between cities), and removes them, yielding three disconnected part-tours. The parts are then reconnected in a different order (by two new edges) such that a new tour is obtained. Using a local search procedure, one would start with an initial solution and try all possible 2-OPT operations until a better permutation has been found. If no improvement is possible, the algorithm stops. Note that 2-OPT defines a neighborhood for each point (tour) in the search space in terms of all the possible new arrangements of three parts of that tour. The size and the structure of such neighborhoods are crucial for the success of these algorithms.

Once a neighborhood in a local search algorithm is defined, we can address the problem of becoming trapped in local optima. By choosing a large neighborhood, local optima become less likely. In the extreme case, one might think of defining the neighborhood of a solution as the set of all other solutions in the entire search space, which by definition would include the globally optimal solution. However, it is obvious that this would lead to neighborhoods that are usually not searchable in an efficient way, as the number of elements in them would be exponential with respect to the given problem size.

Considering how to choose good representations and operators, and hence neighborhoods, in our setting can be done by examining the fitness landscape, defined by the search space *S*, the function *f* to be optimized, and the chosen neighborhood N: $S \rightarrow 2^S$. We can think of the fitness landscape as a graph whose elements of *S* are nodes that have certain values, and with an arc from *x* to *y* if *y* is an element of the neighborhood of *x*, that is, $y \hat{I} N(x)$. Fitness landscapes are often visualized by plotting the surface of fitness values over the search space. Solutions that are neighbors are close to each other, that is, they can be easily reached using the operators from (3) above. Because the fitness function *f* often produces similar values for nearby solutions, one can observe local and global optima in the fitness landscape.

High-Dimensionality Fitness Spaces

Finding the global optimum requires a search algorithm to avoid being trapped in any one of possibly very many local maxima. Recent work done within the context of fitness landscapes defined on genotype spaces suggests that landscapes with extremely high dimensionality have certain features that may

From “Cognitive Search: Evolution, Algorithms, and the Brain,” edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

simplify searching the space of solutions. To illustrate this work, consider the following model. Assume that the search space consists of genotypes each comprising a very large number L of diallelic loci (i.e., positions at which they can have one of two different alleles). Each genotype has L one-step neighbors (single mutants). Let us assign fitnesses randomly and independently to each genotype so that they are equal either to 1 (a viable genotype) or 0 (inviable genotype), with probabilities P and $1 - P$, respectively. In general, viable genotypes will tend to form connected networks—that is, they will be connected by steps of a single mutation. For small values of P, there are two qualitatively different regimes: (a) subcritical, in which all connected components of the genotype network are relatively small (which takes place when $P < P_c$, where P_c is the percolation threshold), and (b) supercritical, in which the majority of viable genotypes are connected in a single giant component, which takes place when $P > P_c$ (Gavrilets and Gravner 1997). A very important, though counterintuitive, feature of this model is that the percolation threshold is approximately the reciprocal of the dimensionality of the genotype space: $P_c \approx 1/L$, and thus P_c is very small if L is large (see Gavrilets 2004; Gavrilets and Gravner 1997). Therefore, increasing the dimensionality of the genotype space, L, while keeping constant the probability of being viable, P, makes the formation of the giant component unavoidable. (Similar findings hold when the model is generalized to use continued fitness values; see Gavrilets and Gravner 1997).

In the literature, the connected networks discussed in the previous paragraph are often referred to as neutral networks, where the word "neutral" means that there is no difference in fitness between the genotypes in the network. In certain applications, small differences in fitness are allowed and the resulting networks are called "nearly neutral." The earlier work on neutral and nearly neutral networks in multidimensional fitness landscapes concentrated exclusively on genotype spaces in which each individual was characterized by a discrete set of genes. However, many features of biological organisms that are actually observable and/or measurable are described by continuously varying variables such as size, weight, color, or concentration. A question of particular biological interest is whether (nearly) neutral networks are as prominent in a continuous phenotype space as they are in the discrete genotype space. Recent results provide an affirmative answer to this question. Specifically, Gravner et al. (2007) have shown that in a simple model of random fitness assignment, viable phenotypes are likely to form a large connected cluster even if their overall frequency is very low, provided the dimensionality of the phenotype space L (i.e., the number of phenotypic characters) is sufficiently large. In fact, the percolation threshold, P_c, for the probability of being viable scales with L as $1/2^L$ and thus decreases much faster than $1/L$, which is characteristic of the analogous discrete genotype space model.

Earlier work on nearly neutral networks was also limited to consideration of the direct relationship between genotype and fitness. Any phenotypic properties that usually mediate this relationship in real biological organisms were

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

neglected. Gravner et al. (2007) studied a novel model in which phenotype is introduced explicitly. In their model, the relationships—both between genotype and phenotype as well as between phenotype and fitness—are of the many-to-one type, so that neutrality is present at both the phenotype and the fitness levels. Moreover, their model results in a correlated fitness landscape in which similar genotypes are more likely to have similar fitnesses. Gravner et al. (2007) showed that phenotypic neutrality and correlation between fitnesses can reduce the percolation threshold, making the formation of percolating networks easier.

Overall, the results of Gravner and colleagues reinforce the previous conclusion (Gavrilets 1997, 2004; Gavrilets and Gravner 1997; Reidys and Stadler 2001, 2002; Reidys et al. 1997) that extensive networks of genotypes with approximately similar fitnesses are a general feature of multidimensional fitness landscapes (both uncorrelated and correlated, as well as in both genotype and phenotype spaces). An important question is whether such concepts could inform internal search over cognitive representations. If so, they would suggest that moving to higher-dimensional search spaces could facilitate internal search by allowing the system to escape from local search optima.

High- and Low-Dimensionality Search

As discussed by Marshall and Neumann (this volume), choosing an appropriate neighborhood representation can make hard computational search problems much easier. Intuitively, one may think that reducing the dimensionality of a search space would make search easier. In machine classification problems, however, appropriately increasing the dimension of the search space (using "kernel methods") can turn a hard nonlinear classification problem into an easy linear one (e.g., Shawe-Taylor and Cristianini 2004), and neural models have been proposed which suggest that brains might also do this (e.g., Huerta et al. 2004). For internal cognitive search, could dynamic adjustment of the dimensionality of the internal space improve search performance? For low-dimensional search landscapes, a well-defined "fitness gradient" at any point in the space exists, but following it can lead a searcher to become trapped in local optima. However, higher-dimensional fitness landscapes have highly connected components (as described in the previous section) with much smaller fitness gradients, allowing neutral diffusion through the entire search space without having to suffer large losses in fitness. The proposal then is that internal search might first search in a low-dimensional internal space, climbing fitness gradients, until a local optimum is reached and no further improvement can be found. This could then be followed by an increase in the dimensionality of the internal space and an episode of neutral diffusion through the space. This would, in turn, be followed by a return to the low-dimensional representation of internal space and a further episode of hill climbing, which may climb

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

a gradient to a new and better local optimum than the one found prior to the preceding episode of neutral diffusion. Several iterations of this process could be used in an attempt to find successive improvements in the quality of local optima discovered.

An approximation of this process can be seen in high-dimensional semantic memory models such as HAL and LSA (Burgess and Lund 2000; Landauer and Dumais 1997). These models capitalize on lexical co-occurrence and acquire word meaning by bootstrapping conceptual representations via the inductive encoding of statistical regularities in language. In the case of HAL, words are represented by vectors, typically with 200–140,000 vector elements, where each element corresponds to another word in the input stream that occurred near the word being represented. The meaning is thus a representation of the contexts in which the word occurred, and input samples can be very large (one billion words has been one of the larger language samples). The vectors are formed by encoding weighted lexical co-occurrence values as a window (typically 5–10 words) moves along the text calculating the vector values for each target word and the words in the window before and after it. Although these models are usually used statically (i.e., the vector values for words are extracted after the model passes through the entire text), they could be used dynamically, in line with the changing dimensionality approach suggested above. Such a model would start small (with about five encoded words and hence five vector dimensions) and add dimensions (and encoded words) as it encounters each unique word. The resulting model would have very sparse dimensionality in that most of the space defined by the dimensions would be unoccupied. Once the model has experienced a large amount of text, dimensionality can be reduced again by retaining the most contextually diverse columns. Regardless of the final number of dimensions, both models can usually undergo a dimensionality reduction to around 200–300 dimensions without losing resolution in their cognitive predictability (e.g., predicting word relatedness, semantic priming, typicality effects, and grammatical and semantic categorization). Both HAL and LSA have been shown to account for various phenomena in the concept acquisition process (Landauer and Dumais 1997; Li et al. 2000) and demonstrate the plausibility of dynamically increasing and decreasing dimensionality of the space, as needed, to represent the language input.

Representations Learned by Humans and Machines

As the foregoing analysis of fitness landscapes attests, a central factor for a search process is how the search space is represented. Relatedly, in the domain of cognitive science there is a history of research showing that representations change as people gain experience during search. Prominent examples are when people learn how to solve a complex problem or acquire a complex skill, such as when one learns how to navigate in a city or learns how to play chess. Studies on expert-novice differences in chess consistently show that one important element that defines chess expertise is whether the person can effectively represent

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

the states of a chessboard to promote inferring the best move. For example, expert chess players are often found to have exceptional memory of chess positions and are able to recall them accurately even after a short (< 5 s) encoding time. Exceptional memory, however, is only found when the chess positions are meaningful (Chase and Simon 1973b). When chess pieces are randomly located on the chessboard, recall accuracy decreases dramatically. This is often taken as evidence that experts have more efficient internal representations of the chess positions that allow them to interpret quickly the functional state of the game. In other words, extensive experience with the search space (possible moves in a chess game) allows experts to reduce the dimensionality of the search space, making their search more efficient than for novices.

Chess playing is also studied extensively in the domain of machine learning. In fact, developing algorithms that can beat human chess players is often considered a major benchmark test for success in the field of artificial intelligence. One common approach is to compute the optimal depth of win (minimum number of moves to win) for a given state, and use this as a fitness function in the search algorithm, based on which the computer selects the "best" move. Finding the best move often requires extensive search in a very large space of possible moves, and it must be done over and over, because the search space changes after each move by the opponent. Nevertheless, rapid advances in machine learning techniques and computational power have led to machines that can beat even the most skilled human chess players. On the other hand, the way a computer plays chess is very different from the way in which a human plays. Specifically, it is believed that the search process is much more efficient for humans than computers in the sense that humans consider vastly fewer moves. The reduction of the search space through experience is often considered the primary reason why cognitive (human) search is more efficient than machine search.

The human ability to develop better representations that facilitate search becomes even clearer in cases where the size of the search space is larger than it is for chess. For example, while machines can beat a human chess player, no machine algorithm has yet been developed to beat expert players of the game of Go—an ancient board game for two players that is noted for being rich in strategy despite its relatively simple rules. Because Go utilizes a much simpler set of rules than chess, the search space becomes much less constrained, thus making it difficult for a machine to search. On the other hand, expert Go players, like expert chess players, can learn more effective representations of the search space by perceptually recognizing "loosely defined" functional states through experience, which practically reduces the dimensions of the search space they use.

As discussed by Fu (this volume), the way that representations and search processes interact is often considered a fundamental aspect of intelligence. The discussion above leads to the perhaps paradoxical conclusion that the amount of search is not necessarily a measure of the amount of intelligence being

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

exhibited by the agent (human, animal, or machine). What makes search intelligent is not that a large number of search steps are required for reaching the target, but that a large amount of search would be required if a requisite level of knowledge were not applied by the cognitive system (Newell and Simon 1976, 1987). While it seems that there is still no deep theoretical explanation for the distinction in performance between human experts and machines, there are three general conclusions that can be based on the observations. First, some part of the human superiority in tasks with a large perceptual component, as in chess or even the traveling salesman problem (MacGregor et al. 2000), can be attributed to the special-purpose, built-in, parallel processing structure of the human perceptual-spatial system. Second, many of the tasks in which humans excel seem to involve a large amount of semantic information. For example, master-level chess players are estimated to have knowledge of approximately 50,000 relevant chess patterns. This suggests that experts can substitute recognition for search (at least partially) because these patterns contain an enormous amount of information that helps the experts to reduce the search space significantly. Finally, there may be a distinction between local and nonlocal use of search knowledge (see Hills and Dukas, this volume). Many chess algorithms tend to use information gathered during the course of search (we refer to this kind of information broadly as "search knowledge") only locally to help make decisions at the specific (or neighboring) node where the information was gathered. Hence, the same facts have to be rediscovered repeatedly at multiple locations in the search space. Humans, however, are good at taking search knowledge "out of context" and generalizing it to apply to a wider range of areas. Thus, if a weakness in a chess position can be traced back to a series of moves that led to it, then the same weakness can be expected in other positions if the same (or similar) series of moves is executed. Indeed, much progress has been made in machine learning in this kind of nonlocal use of knowledge to improve search. Just how (e.g., mechanistically) humans are able to do so is still relatively unknown. However, the importance of choosing the appropriate representations seems to be a key factor that influences search performance.

In summary, we argue that dimension reduction of the search space by experience is one critical characteristic of cognitive search that distinguishes it from formal methods of search developed by machine learning researchers. This type of representational change seems to be the main reason why cognitive search can be more efficient than machine search. More research is needed to uncover how this kind of dimension reduction in the representation of search space is accomplished and what neurocognitive mechanisms are involved.

Built-In and Learned Constraints

The importance of constraints for search is strongly supported by both theoretical and empirical arguments. Classic work on heuristics has shown that

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

efficient search depends on the searcher being able to apply operators that usually bring the searcher closer to its goal. An unbiased or unconstrained searcher will typically be unable to find goals in reasonable amounts of time.

Consider the vast space of possible language grammars. Gold and Chomsky formally showed that there are too many possible grammars to learn a language in a finite amount of time, let alone the two years required by most children, if there were no constraints on what those grammars might look like (Gold and The RAND Corporation 1967; Chomsky 1965). In a related analysis, Wolpert (1996) showed that there is no such thing as a truly general and efficient learning device. Developmental psychologists have argued that children need to have built-in constraints, biases, or implicit assumptions that fit well with their environment (Gelman 1990; Spelke and Kinzler 2007).

One exciting alternative to built-in constraints is that experience with a richly and diversely structured world can allow agents to devise some of the constraints that they will then use to make searching their world for adaptive behaviors more efficient. While some constraints are surely provided by evolution, others can be acquired during an organism's lifetime and are no less powerful for being learned. In fact, acquired constraints have the advantage of being tailored to an individual's own circumstances. For example, early language experience establishes general hypotheses about how stress patterns inform word boundaries (Jusczyk et al. 1999). Children are flexible enough to acquire either the constraints imposed by a stress-timed language (e.g., English) or a syllable-timed language (e.g., Italian), but once the systematicities within a language are imprinted, children are constrained to segment speech streams into words according to these acquired biases. When exposed to new objects, people create new descriptions for the objects' parts and then are constrained to use these descriptions to represent still later objects (Schyns and Rodet 1997). As a final example, Madole and Cohen (1995) describe how 14-month-old children learn part-function correlations that violate real-world events. These correlations cannot be learned by 18-month-old children, which suggest that children younger than this acquire constraints on the types of correlations that they will learn. In all of these cases, constraints are acquired that subsequently influence how people will search for regularities in their environment.

A search system must have strong constraints on the possibilities it will pursue if it wants to find solutions in a practical amount of time, but a considerable amount of flexibility is still needed when a system faces different environments and tasks. This dilemma can be resolved by again making constraints themselves learnable. Kemp et al. (2010a, b) present a quantitative, formal approach to learning constraints. Their hierarchical Bayesian framework describes a method for learning constraints at multiple levels. For example, upon seeing several normal dogs, their system would develop expectancies of various strengths that a new dog will have four legs, that mammals have four legs, and that animals have four legs. Upon seeing a set of both dogs and swans, their system would expect dogs to have four legs, swans to have two legs,

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

and, more generally, all animals of a particular species to have a characteristic number of legs. This latter hypothesis will in turn help the system to quickly find the hypothesis "all beetles have six legs" upon seeing only a single beetle exemplar. Representations at higher levels capture knowledge that supports learning at the next level down. In this manner, constraints can be learned at a higher level that facilitate search for valid inferences at a lower level.

Bayesian approaches are not the only models that can search for constraints for further search. Some neural network models provide working examples of systems that learn new constraint structures because of the inputs provided to them. Bernd Fritzke's (1994) *growing neural gas model* provides a compelling example of this. When inputs are presented, edges are grown between nodes that are close to the input, and new nodes are created if no node is sufficiently close to the input. The result is a graph-based "skeleton" that can aptly accommodate new knowledge because the skeleton was formed exactly in order to accommodate the knowledge. This skeleton-creating approach appears also in "Rethinking Innateness" (Elman et al. 1996), where one of the primary ideas is that the existence of modularity does not implicate innateness. Modules can be learned through the process of systems self-organizing to have increasingly rich and differentiated structure. Computational modeling suggests that the eventual specialization of a neural module often belies its rather general origins (Jacobs et al. 1991). Very general neural differences, such as whether a set of neurons has a little or a lot of overlap in their receptive fields, can lead to large-scale functional differences, such as specializing spontaneously to handle either categorical or continuous judgment tasks or snowballing into "what" versus "where" visual systems (Jacobs and Jordan 1992). Without belaboring the details of these models, there are a sufficient number of examples of constraint-creating mechanisms to believe that systems can achieve both efficient and flexible search routines by learning how to constrain themselves.

Working Memory Constraints

Beyond imprecise or incomplete knowledge about the search space, biological systems face constraints, such as limited working memory capacities, that make the actual calculation (and memorization) of the optimal solution to many kinds of problems impossible. To help overcome these constraints, humans employ a variety of easy-to-compute strategies and heuristics. Moreover, it appears reasonable to assume that humans create internal representations of the problem space (e.g., the environment) that facilitate the search process. Memory representations of large-scale environments are often described as being hierarchically structured with multiple layers of abstraction (e.g., Stevens and Coupe 1978). A possible function of this organization is that it reduces memory and computation costs when searching for paths between (multiple) locations. For example, by using different levels of detail simultaneously (i.e.,

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

by using high resolution only for the current surrounding while using coarser representations for distant locations), search costs and working memory load are reduced.

Planning a novel route through a familiar environment can be conceptualized as searching for a path through state- or search-space from a given start location to a destination. From a computational perspective, such planning tasks become challenging if the environment is large such that many path alternatives are possible and/or if multiple target locations have to be considered (e.g., when solving the TSP). Wiener et al. (2008) recently studied human performance in solving TSPs under conditions that required working memory as opposed to conditions that did not tax memory. When working memory was required, participants performed better if the optimal solution to the TSP required visiting all targets in one region before entering another region. This is best described by a planning (search) algorithm that utilizes an abstraction of the actual problem space to compute an initially coarse solution that is subsequently refined (see also Pizlo et al. 2006). Again, this algorithm operates on a reduced search space that represents an abstraction of the actual search space, thus reducing working memory load and the computational complexity of the problem. We note, however, that search algorithms which operate on abstractions of the actual search space are obviously vulnerable to suboptimal or distorted solutions (e.g., direction judgments; Stevens and Coupe 1978) and may require replanning during actual navigation (Wiener and Mallot 2003). Such additional costs appear to result from the trade-off between the quality of the solution and the constraints inherent to the system.

Working memory can be construed not just as a limitation to be overcome, but also in some cases as a constraint that may serve important functions (Hertwig and Todd 2003). Consistent with this view, Kareev and colleagues have suggested that short-term memory limitations can actually benefit correlation detection (Kareev et al. 1997). They show that smaller sample sizes of environment observations amplify correlations, because both the median and the mode of the sampling distribution of the Pearson correlation exceed the population correlation. As the size of these observed samples is presumably bounded by short-term memory capacity, people with lower short-term memory capacity would be expected to consider smaller samples than those with higher capacity. The result is that the lower-capacity individuals should be more likely to perceive correlations that have been amplified by their more limited short-term memory. Kareev found empirical support for this hypothesis, although some questions have been raised about both the theoretical analysis and the interpretation of the empirical results. For example, small samples lead to high false alarm rates (Juslin and Olsson 2005), and the advantage can only hold if one assumes a decision threshold (Anderson et al. 2005) and relatively large correlations (Kareev 2005). Gaissmaier et al. (2006) suggest that the apparent empirical advantage in detecting correlations for those with lower

From “Cognitive Search: Evolution, Algorithms, and the Brain,” edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

memory capacity may be confounded with an increased likelihood of those with higher capacity to explore.

As another example of possible advantages of memory constraints, Elman (1991) developed a neural network simulation of a task that a young child faces when learning aspects of a language—essentially, searching for a grammar that accounts for the language inputs being heard. For example, the network had to predict number agreement between subject and verb in a sentence, or whether a verb was transitive or intransitive. The network could not learn the full underlying complex grammar when trained from the outset with "adult" language. However, the network succeeded when its limited "short-term memory" (realized as windows on the input sentences) was allowed to grow gradually. Starting with smaller windows helped the network find the statistical regularities across the input sentences. Whether a limited working memory also helps people learn a hierarchical organization of spatial memory (as opposed to semantic or syntactic memory) is an open question.

Constraints on Physical and Cognitive Search Space Topology

Two of the major goals of this Forum were to explore the relationship between search in different domains, especially external "spatial" search and internal "cognitive" search, and to investigate how search strategies scale from low- to high-dimensional environments. In considering the relationship between external search in the environment and internal search over representations of solutions to problems, one might implicitly assume that the primary difference is that external search is low dimensional (typically two or three) and internal search is high dimensional. We propose, however, that this is not the primary distinction between internal and external search. Rather, as we have reviewed, the important difference is that representation of the space for internal search can vary, both in topology and in dimensionality. It is more difficult to change representations in external space. For example, it is relatively difficult for people and ants to build bridges that reduce distances in external space, while in contrast we argue that distances between points in a cognitive search space can be more easily altered by changes to the representation (e.g., by increasing or decreasing the dimensionality of the search space).

In abstract terms, the topology of a search space is defined by the neighborhood function that specifies how points in the search space relate to each other. In external space this has a natural interpretation: on the surface of the Earth, the neighbors of a particular point in space have an intuitive definition, which similarly holds for the three-dimensional spaces inhabited by animals able to fly or swim. Animals searching in these environments must move through spatially neighboring points before they can get to other more distant points; they cannot teleport. Thus, if an animal wishes to search for something (e.g., food or mates at a distant point), it must move through other points to get there and

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

might take the opportunity to search at intermediate points as well along the way. In transferring the concept of external search to internal search, however, one should realize that the spatial structure or topology of an environment is outside the animal's control, while for internal search, the representation of the space, and hence the topology and dimensionality of the space itself, can be changed. This could, in turn, affect the difficulty of a search process in that internal space; hence, a useful representation for an internal search problem might itself be searched for, or selected, by the animal or by evolutionary processes in the "space of possible representations" (Newell and Simon 1976, 1987).

Animals do have some ability to change the dimensionality of their environment. Consider, for example, an ant colony reducing a two-dimensional surface to a network of one-dimensional pheromone trails and manipulating the nature of that network to facilitate navigation (e.g., Jackson et al. 2004), or a terrestrial animal that increases the dimensionality of its environment by acquiring the ability to fly. Animals may also directly adapt the dimensionality of their search to achieve some objective; for example, switching between trail following and more exploratory behavior in the case of ants (Edelstein-Keshet et al. 1995) or changing between local terrestrial search and long-distance flights between areas in the case of a bird or other flying animal (Amano and Katayama 2009). Nonetheless, an animal's ability to manipulate the external space in which it searches is limited by dimensionality and the basic laws of physics. In contrast, internal search spaces should be subject to much less restriction, both in terms of dimensionality and topology.

Social Search

Newell and Simon focused on the intelligence of individuals, but groups also need to act intelligently to search for solutions. A key distinction is between group search and individual search with a social component (for an extended discussion, see Lazer and Bernstein, this volume). Group search involves group-level payoff, whereas individual search involves individual-level payoff. In both cases, one can still examine the collective implications of individual behavior, but the presence of a group payoff potentially reduces the conflict of interest among individuals. Individual success is some function of collective and individual components, and the relative magnitude of these components. At one end of the spectrum, individual success is purely a function of group success (social insects may be closer to this end). In other group systems, individual success might be empirically separable from group success, or there might be no group component to individual success whatsoever. Thus, for example, there might be individual benefits to putting less effort into foraging, even though the group (and individual) also gains benefits from

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

finding resources. The general conundrum is that exploration is individually costly but offers benefits to the collective (more on this below), so the potential risk for the group is underinvestment in exploration. How then to achieve group search? From a behavioral ecology perspective, the conditions needed for group selection to emerge at the genetic level (e.g., very low levels of genetic mixing) are quite narrow and unlikely to have characterized humans or human predecessors. Instead, culture might be a potential avenue for group selection because of the speed of cultural relative to genetic change.

Communication in Social Search

The individual-group dimension is actually part of the general question of what the structure of payoff interdependence among actors is. While the dominant idea of foraging is that there is an exhaustible resource, creating a potential conflict of interest among actors, there are many examples of other types of interdependence. Most notably, there is an array of scenarios where agents benefit from the presence of other individuals. For example, one explanation for the existence of cities is the ease with which individuals can communicate information (Glaeser et al. 1992).

Another important dimension is how advertent and inadvertent communication helps coordinate search. Communication, most critically, facilitates exploitation across agents. Agent A discovers resource X, communicates that to Agent B, which exploits resource X. Communication may thus facilitate efficient exploitation of resources, but may also create the social dilemma of overrewarding exploiting agents compared to exploring agents. If agents explore and what is found remains private until the agent shares the information, then reciprocity may be needed to resolve the collective dilemma. If agents explore and what is found is clearly visible to all, and it is not possible to exclude other agents from consuming the good, then an under-investment in exploration will occur. In particular cases, variation in visibility (e.g., some solutions may not be visible or possible to copy, while others are) may occur, which would create a bias toward search for nonvisible resources.

Communication may also be important in efficient exploration. Organized search may be more efficient than uncoordinated search. For example, a group search for missing keys can be more efficient if the searchers look in mutually exclusive sets of rooms—but if Agents A and B have no way of communicating which rooms they have inspected, there is a risk that they both search the same room. In addition, copying behavior may allow for more efficient collective search by focusing search on promising areas of the solution space (i.e., effective exploration sometimes requires effective exploitation).

Often, a contrast is drawn between the emergent patterns of self-organized groups and groups that are driven top-down by a leader, rule system, or

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

hierarchical structure (Resnick 1994). What this rhetorical antithesis misses is that self-organized groups do elect leaders, form rule systems, and institute hierarchies (akin to changing the search space representation as described earlier). Most groups that follow rules are typically self-organized, and the rule systems themselves are self-organized. The rules are the tangible products of courts, parliaments, congresses, and governments at city, regional, national, and global levels. For example, in the absence of an existing governmental structure to regulate lobster harvesting effectively, the harvesters themselves created a structure (Acheson 2003). Rules and norms (their less explicit cousin) are complex systems in their own right, no less so than beehives or traffic jams. They do not exist on their own, but rather depend upon supporting structures for their continuation. They require legal and governmental systems to be created, changed, and eliminated (Ostrom et al. 2003).They require monitoring systems (e.g., police) to insure adherence and sanctioning systems (e.g., jails) to punish discovered rule violations. Originally unorganized groups will propose, vote upon, and live under rule, monitoring, and sanction systems that they construct themselves (Janssen et al. 2008; Samuelson and Messick 1995). In this manner, groups that face scarce resources are often importantly not simple decentralized systems, but rather decentralized systems that spontaneously create rule systems that are themselves decentralized.

Humans are not alone in adaptively creating organization structures that help them achieve their goals. Some ant species tune their level of egalitarianism to the level of informational uncertainty of individuals within the colony (Sueur et al. 2011). When individuals have little uncertainty about the relative advantages of different resources in their environment, they adopt more despotic decision regimes in which group choices are controlled by relatively few individuals (for a related point, see Pierce and White 1999). When informational uncertainty is low, or when decisions must be made quickly, there are benefits for social search processes that concentrate effective voting power in relatively few individuals. As informational uncertainty increases, so does the importance of pooling information across many individuals. In related work, bee swarms searching for new nesting sites have been aptly modeled as a population of agents that accumulate evidence for alternative choices (Marshall et al. 2009; Seeley et al. 2012). Assuming that the colony has adapted to achieve at least a certain level of accuracy at discovering the best available nest site, this accumulation process involving many individuals minimizes search time. Similarly for human groups, when the complexity of a problem space is low, centralized networks in which a single individual communicates with others are effective in a manner that no longer is found as problem complexity increases (Leavitt 1962); but distributed networks become important as the ruggedness of a problem space increases (Lazer and Friedman 2007).

From “Cognitive Search: Evolution, Algorithms, and the Brain,” edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

Conclusion

As Newell and Simon famously eschewed disciplinary boundaries, one can only imagine how pleased they would be to see how search informs and connects the cognitive, biological, and social sciences today. In this chapter, we have described the benefits of restructuring search spaces and internal representations so as to make searches more efficient. Whereas Newell and Simon focused on the application of heuristics to fixed and well-defined search spaces, biological and social systems often engage in higher-level searching for more effective representations to make their lower-level searches more effective. This can be achieved by either increasing or decreasing the dimensionality of internal representations, or by restructuring the representations altogether.

As clever as they were, Newell and Simon could not be expected to predict perfectly the developments in science 35 years later. For example, Newell and Simon (1976, 1987) thought that mimicking the way people play chess was the most promising way forward for chess programs. At the time of their Turing award, such programs had only just begun "to compete with serious amateurs." They believed the route computers would take to beat the best human players would be to buttress heuristic search with knowledge. In the end, although heuristics certainly played a role in the computer victory over people, Hitech and its successor Deep Blue depended more on the "massive" search of game trees than Newell and Simon had imagined; a triumph of Moore's law regarding exponentially increasing computer processing power. In their words: "It's fun to be wrong" (Newell and Simon 1987:316). Although they admittedly missed the mark with respect to the extent that human-inspired heuristics would solve the problems of artificial intelligence, their take on the key role of heuristic search for human intelligence does appear to have been largely substantiated. As we have reviewed here, human intelligence depends on constraining search in a variety of ways. It's also fun to be right.

Acknowledgments

We thank Iain Couzin and John M. C. Hutchinson for helpful comments.

From "Cognitive Search: Evolution, Algorithms, and the Brain," edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.

From “Cognitive Search: Evolution, Algorithms, and the Brain,” edited by Peter M. Todd, Thomas T. Hills, and Trevor W. Robbins. 2012. Strüngmann Forum Report, vol. 9, J. Lupp, series ed. Cambridge, MA: MIT Press. ISBN 978-0-262-01809-8.