Investigating the role of the visual system in solving the traveling salesperson problem

Zahra Sajedinia (zsajedin@purdue.edu)
Department of Psychological Sciences, 703 Third Street
West Lafayette, IN 47907 USA

Zygmunt Pizlo (zpizlo@uci.edu)
Department of Cognitive Sciences, University of California
Irvine, CA 92697 USA

Sébastien Hélie (shelie@purdue.edu)
Department of Psychological Sciences, 703 Third Street
West Lafayette, IN 47907 USA

Abstract

This article used an empirical experiment and a computational model to test the hypothesis that humans rely on the visual system to solve the traveling salesperson problem (TSP). We tested two consequences of this hypothesis: (1) humans should perform better on Euclidean TSP than not–Euclidean TSP; (2) a model of the visual system should account for performance in Euclidean TSP. Participants were asked to solve Euclidean or not–Euclidean TSP, and a pyramid model of the visual system was used to solve the same tours as the humans. The results show that deviations from the optimal tour were smaller in Euclidean problems than in not–Euclidean problems, and the fit of the pyramid model to human performance was worse on not–Euclidean problems then on Euclidean problems. These results suggest that participants solve Euclidean problems with the visual system, but that other mechanisms are needed to succesfully solve non–visual problems.

Keywords:

Problem Solving; Visual Processing; Traveling Salesperson Problem; Pyramid Model

Introduction

A problem is a situation in which an agent seeks to attain a given goal without knowing how to achieve it. Humans solve problems every day. Example problems include winning at tic–tac–toe or winning a battle, air traffic control, control of an uninhabited vehicle, getting to checkmate in chess, visually–guided navigation, proving a logic theorem, solving math and physics problems, cracking the enigma code, or formulating a new scientific theory. Some problems are more visual, such as planning a tour around a grocery store, while others are more abstract, such as proving a theorem using predicate logic. In this conference article, we focus on the Traveling Salesperson Problem (TSP), a well–known optimization problem. In the TSP, a set of points is presented to participants. Each point represents a city, and the goal is to find the shortest possible route that visits all the cities exactly once, and returning to the starting city. We refer to this route as a TSP tour. The TSP has high relevance since it (1) has an important visual component (i.e., cities or points are spatially laid out on a map) and (2) it has important real–life application in many areas such as logistics, transportation, and shipping.

TSP has been studied extensively by cognitive scientists to reveal the underlying processes in human problem solving (van Rooij et al., 2006; Chronicle et al., 2008; Dry et al., 2006; MacGregor, 2013). One reason that makes the TSP an interesting problem for cognitive scientists is that the problem space of the TSP is very large. Even for solving a 16 city TSP, there are 6×10^{11} possible solutions, which is more than the number of neurons in the human brain (Azevedo et al., 2009). Also, the TSP is proven to be computationally NP–hard, meaning that there is no algorithm that can find an exact optimal solution for the TSP in polynomial time (Pizlo & Stefanov, 2013).

Human working memory can only store and manipulate a few items at a time and cannot make more than a few comparisons at a time (Pizlo & Stefanov, 2013). Yet, even with these severe limitations in memory and processing power, humans are able to solve the TSP near optimally in approximately linear time (MacGregor & Chu, 2011; Pizlo et al., 2006). How can humans with these limitations be able to solve the TSP fast and near optimally? What cognitive systems and processes have evolved to solve the TSP in the human brain?

Goals and Hypotheses

Pizlo and colleagues have argued that the TSP is solved by parallel processes in a pyramid–like hierarchical architecture of the visual system (Graham et al., 2000; Pizlo et al., 2006; Pizlo & Stefanov, 2013). The assumption that humans solve the TSP visually has important implications on the types of problems that can be solved. The human visual system has evolved in Euclidean space, so the visual system likely assumes a Euclidean cost function when solving optimization problems. As a result, performance in optimization problems with not–Euclidean or non–metric cost functions might be impaired.

To test this hypothesis, we designed a TSP experiment where participants solved either a regular (Euclidean) or not–Euclidean TSP. The participant's data was then compared with tours produced by a well–known computational model of the visual system, namely the pyramid model (Adelson et al., 1984; Pizlo et al., 1995). According to our hypothesis, human participants should perform well in the Euclidean version of the TSP but not in the not–Euclidean version of the TSP. Further, the pyramid model should provide a good account of participant TSP tours in the Euclidean TSP but not in the not–Euclidean TSP. These results would support the hypothesis that participants are solving the regular TSP using the visual system, but not the not–Euclidean TSP. Further, the compensatory mechanisms used to solve the not–Eucliean TSP are not as efficient as the visual system at solving optimization problems.

Method

The first aim of this study was to explore how humans perform in different conditions of the TSP (i.e., Euclidean and not–Euclidean). The second aim was to explore how human performance is compatible with the visual pyramid model. The experiment and model are described in turns.

Participants

Ninety–one Purdue undergraduate students participated in the experiment for course credit. Participants were randomly assigned to one of three conditions: Single–color ($n = 36$), Colored–with–no–switch–cost ($n = 28$), and Colored–with–switch–cost ($n = 27$).

Apparatus and Stimuli

The stimuli were 30 maps each generated by putting 50 randomly scattered cities (points) in a $900px \times 900px$ display. The minimum distance between two cities was set to $50px$ to prevent overlapping points. The resulting set of 30 maps was used to create two different stimulus sets. In the first stimulus set, all cities were colored red. This stimulus set is referred as containing *single–color maps* (See Figure 1a). In the second stimulus set, half of the cities (points) were randomly selected and colored red. The remaining cities (points) were colored blue. This stimulus set is referred as containing *colored maps* (See Figure 1b).

The experiment was run on a regular PC. Stimuli were displayed in a 21–inch monitor (1,920 × 1,080 resolution). Participants responded by clicking on the city (point) that they wanted to visit next using a regular computer mouse. After each mouse click, a dark blue edge was drawn between the last visited city and the city that was clicked in the current trial. The order of the city visited was recorded.

Procedure

Each participant solved all 30 maps in one of three conditions. (1) *Single–color (Euclidean)*: This was a typical TSP experiment. The first stimulus set was used (i.e., single–color maps). Participants were asked to find the shortest TSP tour on each map, one map at a time. The cost between cities was Euclidean (i.e., the distance on the screen). No feedback was provided. (2) *Colored–with–switch–cost (not–Euclidean)*: The second set of stimuli was used (i.e., colored maps). In this condition, the cost between two points was not always Euclidean. Specifically, when travelling from a blue city to a red city (or vice–versa), the calculated distance (cost) was twice the distance on the screen. Otherwise, when travelling between two cities of the same color, the distance was as seen on the screen. Note that this arrangement can break the triangle inequality and make the cost non–metric. (3) *Colored–with–no–switch–cost (control)*: Similar to (2), this condition used the second set of stimuli (i.e., colored maps). However, the distance between two points was always the distance on the screen, so the colors could be ignored. This condition was designed to control for possible grouping effects that could be created by having cities of two different colors. In all conditions the experimenter explained the cost structure to the participants (as described above) and instructed them to find the tour with the smallest cost for each map.

Pyramid Model

A pyramidal architecture refers to multiple representations of the input data, with different representations having different scales and resolutions. In vision, the input data is the retinal image and the first layer is represented by the retinal ganglion cells. Each ganglion cell receives information from a particular region of the retina called the cell's receptive field. Receptive fields of different cells partially overlap. In the second layer of the pyramid, each "parent" cell receives input from several "child" cells. In the third layer, each "grandparent" cell receives input from several of its children. This process continues until a single cell on the top of the pyramid can "see" the entire image. Cells at lower layers can see small parts of the retinal image but they can process the information with high spatial resolution. Cells in higher layers can see larger parts of the retinal image but with lower resolution. More generally, cells in higher layers can handle only some statistical information about their receptive fields. The mean value of some property, like intensity, speed, contrast and so on, is the simplest example.

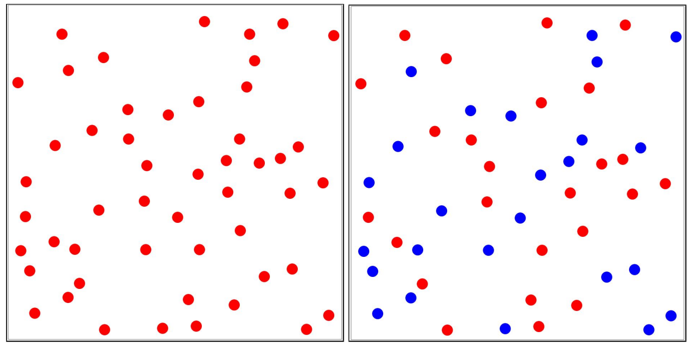
We implemented a Pyramid model adapted from Pizlo et al. (2006). The algorithm is presented in Figure 2. We used Python for our implementation. Inputs to the model were the maps that the participants in the experiment had solved. Because we hypothesized that the visual system evolved in a Euclidean world, the distances between the cities were considered Euclidean in all three conditions. This corresponds to the visual system not being able to process not–Euclidean distances.

Finding Optimal Tours

We used NEOS server of Concorde TSP solver to find the optimal TSP tours for each map. Concorde is one of the best exact TSP solvers currently available. It is available freely for academic use: https://neos-server.org/neos/solvers/co:concorde/TSP.html.

Results

One participant in the Single–color condition had tours that were three standard deviations longer then the condition



(a) Single–color map (b) Colored map

Figure 1: Example maps used in the experiment.

mean. All tours produced by this participants were not included in the following analyses.

Human Performance

Figure 3 shows typical example solutions produced by participants in each condition. As can be seen, the colored–with–switch–cost tour was qualitatively different from those obtained with the single–color and with colored–with–no–switch–cost conditions. Specifically, the not–Euclidean condition included a number of path crossings, which would be suboptimal in Euclidean space (but could be optimal in not–Euclidean space). These crossings were not observed when colors were present without a switch cost.

To quantify the participant performances, the error (i.e., deviation from optimal) was calculated for each map:

$$error_{ji} = \frac{(S_{ji} - O_i)}{O_i} \quad (1)$$

where $error_{ji}$ is the error or participant j on map i, S_{ji} is the length of the tour produced by participant j on map i, and O_i represents the length of the optimal tour for map i .

Table 1 presents the mean error in each condition. As can be seen, the single–color error was 12.6% and the colored–with–no–switch–cost (Euclidean) error was 12.7%, which is almost half of the error observed in the colored–with–switch–cost (not–Euclidean) TSP condition. This shows that participants perform well in Euclidean space but struggle in not–Euclidean space. Also, participants were able to ignore the irrelevant color and the longer tours obtained in the not–Euclidean condition were not caused by a perceptual effect of the city colors. Hence, larger errors for the not–Euclidean condition were not the result of unwanted color grouping effects.

To investigate if the observed differences were statistically significant, we performed Holm–corrected pairwise comparisons t–tests for all three conditions. Error in the not–Euclidean condition significantly differed from error in the single–color ($t(60) = 6.10, p < .0001$) and error in the color–with–no–switch–cost ($t(53) = 5.63, p < .0001$) conditions. The two Euclidean conditions did not differ from each other ($t(61) < 1, n.s.$).

Table 1: Mean participant error in each condition

Condition	Error
Single–color	12.6%
Colored–with–no–switch–cost	12.7%
Colored–with–switch–cost	20.7%

The results show that the errors for the single–color and control conditions were not statistically different. However, the colored–with–switch–cost condition differed from the other two conditions. These statistical differences clearly show that participants' performances were highly dependent on the problem being Euclidean or not–Euclidean, and support the hypothesis that the visual system may assume a Euclidean cost function in solving the TSP.

The performance of the Pyramid model

In Table 2, we compared the Pyramid model generated tours with optimal tours. As can be seen, the error is 14.2% for both Euclidean conditions (single and color), and it increased to 34.5% for the not–Euclidean condition. As expected, the model error was similar to humans in the Euclidean conditions. The RMSD was 4.6% in the single–color condition and 4.5% in the color–with–no–switch–cost condition. However, the model provided a poor fit of human performance in the not–Euclidean condition (RMSD = 15.0%). Assuming that the Pyramid model is an adequate model of human vision, this result suggest that participants solving the Euclidean TSP used the visual system (good model fit), but not the not–Euclidean TSP (poor model fit). Since the participants were doing better than the model in the not–Euclidean condition, this result also suggest that participants may have access to a separate (compensatory) mechanism to attempt to solve not–Euclidean TSP. The pyramid model, in contrast, was purely a model of the visual system and could only deal with Euclidean spaces.

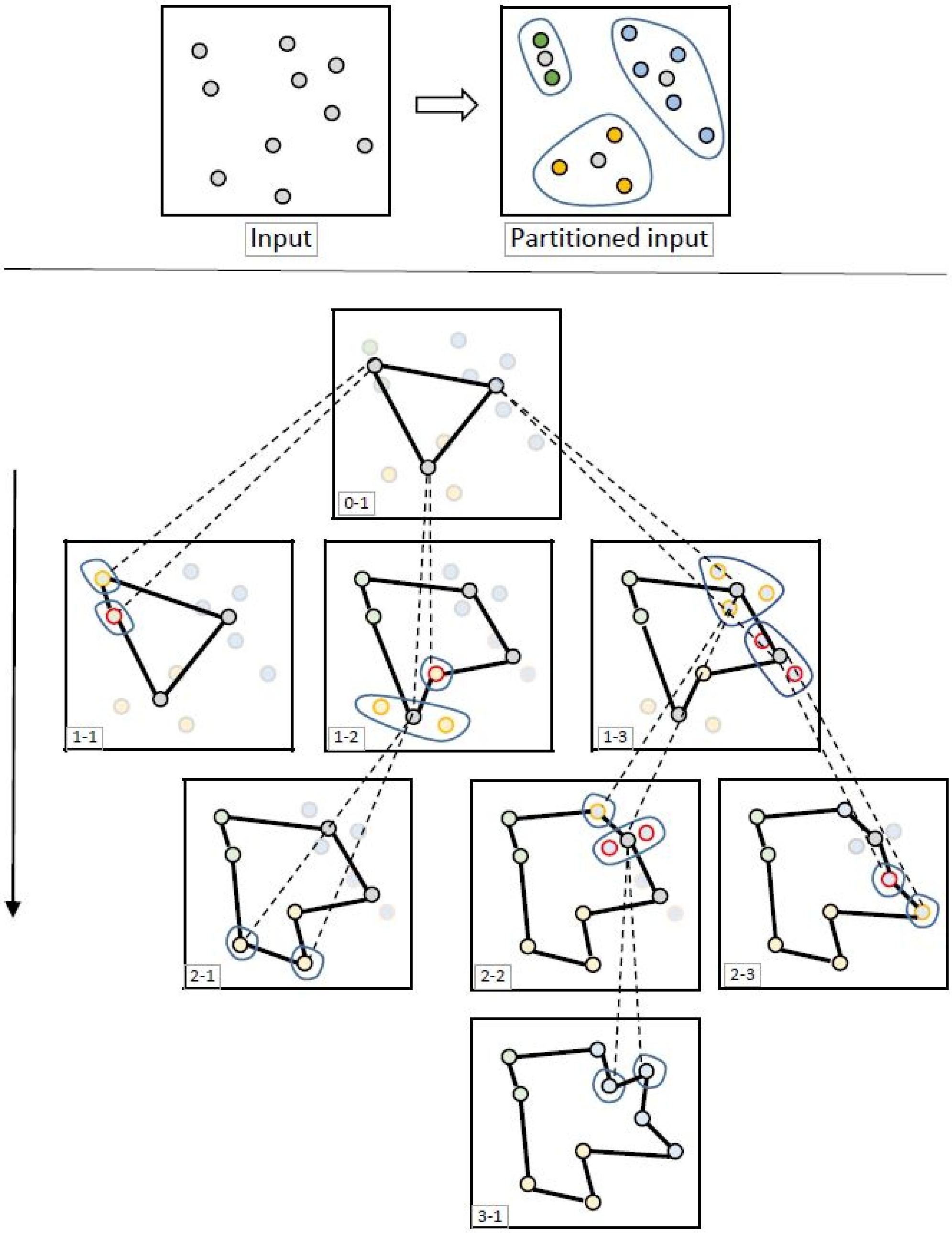
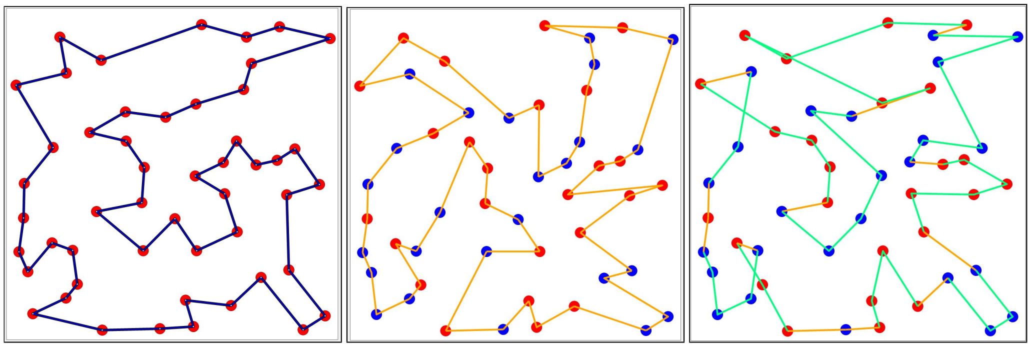


Figure 2: A representation of the Pyramid model. The top row shows the input as it gets partitioned into k clusters (by using a clustering algorithm, such as k–means). In this example, $k = 3$. Next, the pyramid is built. The root of the Pyramid (0–1) is the TSP solution for the centers of the clusters for the partitioned input. The solution for this TSP at the root is trivial because there are only three points and all three points are connected to each other. In the next level of the pyramid (level 1), each cluster is considered separately and recursively repeats the clustering until there is only one point (or city) in each cluster. For example, (1–1) shows the partition of the top–left cluster into k clusters (if the number of points is smaller than k, then $k-1$ is used, here $k = 2$), and a TSP solution for this cluster is found. Since, there were only two points, the solution is trivial, and the two points were connected to each other. Then by brute–force (considering all possibilities), the incoming and outgoing edges are connected to this cluster to obtain the shortest edges. The model then moves to the next cluster (1–2) and repeats the same procedure, until there is no non–visited cluster.



(a) single–color (b) colored–with–no–switch–cost (c) colored–with–switch–cost

Figure 3: Sample tours produced by participants in each condition.

Table 2: The error of the Pyramid model.

Condition	Error
Single–color	14.2%
Colored–with–no–switch–cost	14.2%
Colored–with–switch–cost	34.5%

Discussion

This article used an empirical experiment and a computational model to test the hypothesis that humans solve the TSP by assuming an Euclidean cost function. This assumption follows from the TSP being solved visually, and the visual system having evolved in an Euclidean world. We specifically tested two consequences of this hypothesis, namely that humans would perform better on Euclidean TSP than not–Euclidean TSP and that a model of the visual system could account for performance in Euclidean TSP. Participants were asked to solve the TSP in three conditions, two Euclidean and one not–Euclidean. A pyramid model of the visual system was used to solve the same tours as humans. The results show that the deviations from the optimnal tours were almost twice as small in Euclidean problems than in not–Euclidean problems, and the fit of the pyramid model to human performance was three times worse on not–Euclidean problems than on Euclidean problems.

Relevance for Problem Solving Research

Some problems are visual, like TSP on a Euclidean plane or visual navigation, but other problems may not have an obvious visual representation. Algebra problems, first order logic, and chess are examples. Logic is not visual, but set theory, with Venn diagrams, provides a visual version for at least some logical problems. However, not all problems are amenable to a useful visual representation. In these cases, the massively parallel nature of the visual system is no longer sufficient: problems need to be solved sequentially. One possibility is to use reinforcement learning (Sutton & Barto, 1998). In this framework, the agent is a sequential decision–making system and the environment is another system evaluating the distance between the current problem state and the goal state (Dandurand et al., 2012). In visual cases, the environment could be the visual system with geodesic estimates. In more abstract cases, the environment could be a meta–cognitive system used to evaluate states and rewards. Regardless of how the environment is implemented, actions are selected in each state by using a policy. The policy numerically describes the desirability of each action in each state. The goal of reinforcement learning is to find a policy that maximizes the return, which is the sum of all future rewards, until the problem is solved. However, any sequential system attempting to solve a NP–hard problem, such as the TSP, will quickly be overwhelmed by complexity. This could explain why human participants did better than the pyramid model in not–Euclidean TSP but did not do as well as in the Euclidean problems.

Future Work and Limitations

Future work can be directed in two ways. First, we can further test the theory of the engagement of the visual system in solving the TSP. It can be done by studying whether human performance is compatible with other characteristics of the visual system such as its limited ability to learn. The second direction is proposing a more complete model of human problem solving. Implementing a dual–system model of problem solving, including both a parallel visual module and a sequential decision–making module can be a promising direction. Tentatively, using a reinforcement learning agent for sequential decision–making would allow for learning in problems that cannot be solved visually. This possible dissociation in learning ability for visual and non–visual problems may allow for optimizing the way we represent and solve problems. Future work should be devoted to implementing and testing such a model.

Acknowledgement

This research was supported in part by the National Eye Institute (award #1R01EY024666-01 to PZ) and the National Science Foundation (award #1662230 to SH).

References

Adelson, E. H., Anderson, C. H., Bergen, J. R., Burt, P. J., & Ogden, J. M. (1984). Pyramid methods in image processing. *RCA Engineer*, *29*, 33–41.

Azevedo, F. A., Carvalho, L. R., Grinberg, L. T., Farfel, J. M., Ferretti, R. E., Leite, R. E., ... others (2009). Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain. *Journal of Comparative Neurology*, *513*(5), 532–541.

Chronicle, E. P., MacGregor, J. N., Lee, M., Ormerod, T. C., & Hughes, P. (2008). Individual differences in optimization problem solving: Reconciling conflicting results. *The Journal of Problem Solving*, *2*(1), 4.

Dandurand, F., Shultz, T. R., & A, R. (2012). ncluding cognitive biases and distance-based rewards in a connectionist model of complex problem solving. *Neural Networks*, *25*, 41–56.

Dry, M., Lee, M. D., Vickers, D., & Hughes, P. (2006). Human performance on visually presented traveling salesperson problems with varying numbers of nodes. *The Journal of Problem Solving*, *1*(1), 4.

Graham, S. M., Joshi, A., & Pizlo, Z. (2000). The traveling salesman problem: A hierarchical model. *Memory & Cognition*, *28*, 1191–1204.

MacGregor, J. N. (2013). Effects of cluster location on human performance on the traveling salesperson problem. *The Journal of Problem Solving*, *5*(2), 3.

MacGregor, J. N., & Chu, Y. (2011). Human performance on the traveling salesman and related problems: A review. *The Journal of Problem Solving*, *3*(2), 2.

Pizlo, Z., Rosenfeld, A., & Epelboim, J. (1995). An exponential pyramid model of the time course of size processing. *Vision Research*, *35*, 1089–1107.

Pizlo, Z., & Stefanov, E. (2013). Solving large problems with a small working memory. *The Journal of Problem Solving*, *6*(1), 5.

Pizlo, Z., Stefanov, E., Saalweachter, J., Li, Z., Haxhimusa, Y., & Kropatsch, W. G. (2006). Traveling salesman problem: A foveating pyramid model. *The Journal of Problem Solving*, *1*(1), 8.

Sutton, R., & Barto, A. (1998). *Reinforcement Learning: An Introduction*. MIT Press.

van Rooij, I., Schactman, A., Kadlec, H., & Stege, U. (2006). Perceptual or analytical processing? evidence from children's and adult's performance on the euclidean traveling salesperson problem. *The Journal of Problem Solving*, *1*(1), 6.