REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE 31-08-2009

2. REPORT TYPE Final

3. DATES COVERED May 1, 2006 – May 31 2009

4. TITLE AND SUBTITLE Human Problem Solving: The Complete Model of the Traveling Salesman Problem

5a. CONTRACT NUMBER FA9550-06-1-0161

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Zygmunt Pizlo

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Purdue University, West Lafayette, IN

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-OSR-VA-TR-2012-0149

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
The Traveling Salesman Problem (TSP) is a classical combinatorial optimization problem that has potential for application in several fields, specifically, Cognitive Psychology, Operations Research, and Artificial Intelligence. The TSP refers to finding the shortest tour of a number of cities. Research supported by this grant has shown that humans provide near-optimal solutions very quickly despite the fact that this problem is generally considered to be "computationally intractable". This was demonstrated both for two-dimensional and three-dimensional versions of the TSP that were studied in both real and virtual environments. A new computational model was developed that accounts for all of these results. This model is the first to be able to do this. It is based on a graph–pyramid architecture resembling the established architecture of the human visual system. Furthermore, this model, which was tested under various levels of spatial uncertainty, was found to be quite robust. Finally, a limited-memory version of the model was also developed. It solves combinatorial optimization problems characterized by arbitrarily large search spaces. This development may lead to a new computational theory that explains how humans solve a variety of complex problems such as those found in mathematics, physics, or chess playing.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (include area code)

20120918108

1. Introduction.

Human ability to solve several versions of the Traveling Salesman Problem (TSP) was tested and modeled. TSP refers to the task of finding the shortest tour of a number of cities. Figure 1 shows 6 points and the shortest tour of these points. The total number of tours for this problem is 60. Note that the number of tours grows very quickly with the number of cities. Specifically, for N cities, the number of all different tours is (N-1)!/2. For example, the number of tours in a 16-city TSP is about 10^{12}, which is equal to the number of neurons in the human brain. Because the number of tours grows exponentially with the problem size, TSP belongs to the class of NP hard problems which refers to the fact that finding the shortest tour may require, in the worst case, checking all tours. This is impractical. Therefore, NP hard problems are called intractable.

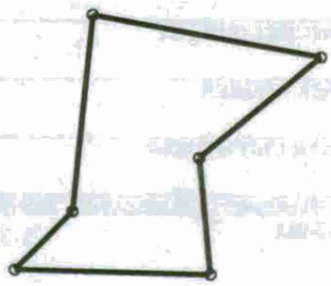


Figure 1. TSP problem with 6 cities.

The list of goals that were accomplished in this project is as follows:

(i) formulation of a new model for solving 2D TSP with obstacles.
(ii) testing subjects in real settings and in virtual reality on 2D and 3D problems
(iii) formulation of a graph-pyramid (MST) model and testing it with 2D and 3D problems
(iv) exploring implications for other cognitive tasks: visual navigation, figure-ground organization
(v) analysis of the role of uncertainty about the positions of the cities in the quality of TSP solution
(vi) formulation of a limited-memory TSP model

The results accomplished for each goal are described next in separate sections.

2. A model for solving 2D TSP with obstacles.

Our first model for solving TSP problems in the presence of obstacles, like that in Figure 2, was a modification of our model for solving 2D Euclidean TSP problems (Saalweachter & Pizlo, 2008). Specifically, the model first solved the problem without obstacles by using the bisection foveating pyramid of Pizlo et al., 2006), and then modified the tour by incorporating the information about where the obstacles are. This was done by verifying whether a given edge in the solution path crosses at least one obstacle. If it does, then a path around the obstacles was found. This path was found by first building the visibility graph and then by applying Dijkstra algorithm for finding the shortest path in a graph. The new model for TSP with obstacles was different in two ways: it used a graph pyramid representation based on Minimum Spanning Tree and it used the Fast Marching Algorithm for finding the shortest paths around obstacles.

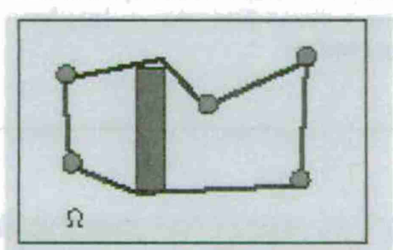


Figure 2. A problem with obstacles with a superimposed tour.

An example of a 50 city problem with obstacles solved by the new model is shown in Figure 3. Note that this algorithm can be applied to any problem, as long as that problem search space can be represented by a graph with known edge lengths. This includes a large collection of problems such as the 15-puzzle, mathematics and physics problems.

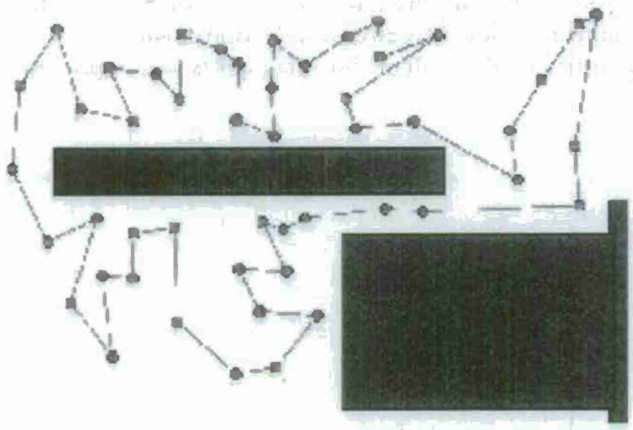


Figure 3. A problem with obstacles solved by the MST pyramid model, which used the fast marching algorithm.

3. Human performance in 2D and 3D real and virtual settings.

TSP is conventionally defined as a problem on a 2D Euclidean plane. Specifically, the "cities" are placed on a plane, their positions are known accurately and the distances between the cities are specified by the Pythagorean formula. The only problem for a problem solver is to determine the optimal or nearly-optimal tour. This is how all experiments with human subjects were conducted in the past. The subject was shown the TSP problem on a computer screen and the screen was orthogonal to the observer's line of sight (see Fig. 3). Under these conditions, the retinal image in the observer's eye is a scaled copy of the image on the computer screen. It follows that an optimal tour for the problem on the observer's retina is also an optimal tour for the problem shown on the computer screen. Furthermore, a suboptimal tour with a relative error ε for the problem on the observer's retina will be a suboptimal tour with the same error for the problem on the computer screen (the relative error is defined as the difference between the length of the tour produced by the subject and the length of the optimal tour, normalized to the latter).

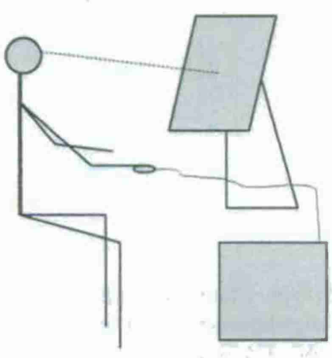


Figure 3. A typical experimental setting for testing human subjects in TSP.

However, this laboratory setting does not generalize to real-life situations. In real life, the "cities" are not necessarily restricted to a plane. Instead, they may reside on a 3D surface or in a 3D volume and their distances are almost never known accurately. Furthermore, the problem is likely to occupy a large part of the observer's field of view, rather than the central 20 deg, or so, as in the case of experiments where the problems are shown on a computer screen.

Consider the task of collecting tennis balls on a tennis court (see Fig. 4). The TSP problem occupies most of the observer's field of view. In fact, once the observer is in the center of the problem space, some parts of the problem are outside of her visual field. She cannot see the whole problem at one glance. Will the observer be able to integrate the visual information across several fixations? Furthermore, the observer cannot solve the problem on her retina because the retinal image is a perspective transformation of the original problem. In order to compute the actual distances on the tennis court, the observer has to perform backprojection of the retinal image. How well can the observer estimate the slant of the floor relative to her eye? Note that even small errors in this estimation will lead to large errors in the reconstructed distances on the floor. For example, assume that the actual slant is 80 deg (as was the case in our experiments), and that the slant is underestimated by 10 deg (this magnitude of error is quite realistic). This error in estimating slant of perspective projection will lead to up to 50% errors in estimating distances on the floor! Such big errors will imply grossly suboptimal TSP tours.

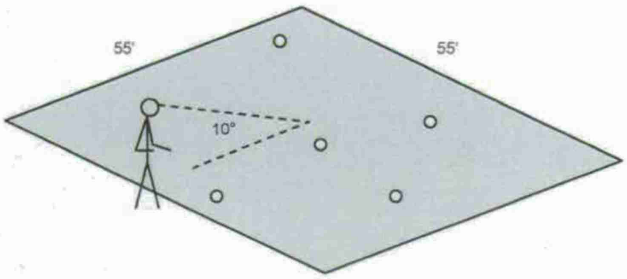


Figure 4. TSP on a tennis court. The subject's retinal image is a perspective projection of the TSP problem with an average slant of 80 deg.

In order to evaluate the usefulness of virtual reality tools in studying human perception and problem solving, the tennis court experiment was replicated by using Purdue "Cave" (see Fig. 5). The Cave consists of four screens (three walls and the floor). The observer can move within the Cave by walking. The head tracker records the position of the observer and updates the visual information, in order to adequately simulate the change of visual scene. The observer wears goggles that provide her with binocular vision.

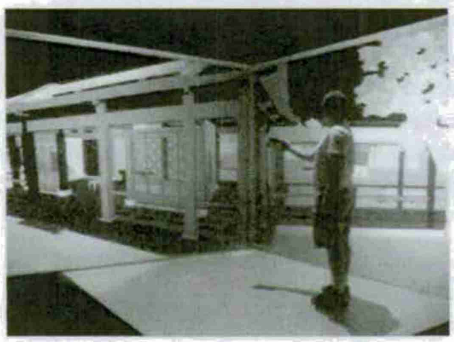


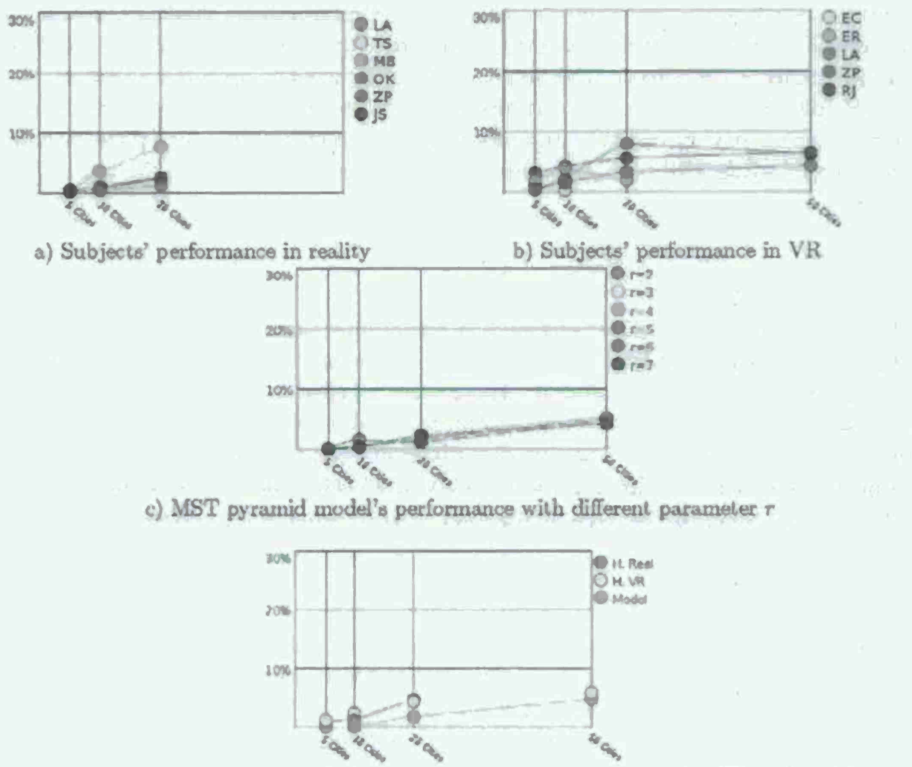
Figure 5. Purdue "Cave" was used to simulate the real settings in a virtual reality environment.

3.1 Experiment on 2D real and virtual TSP.

The size of the area within which the "cities" (tennis balls) were placed was 55 by 55 foot. The number of cities, which represents the problem size n, was 5, 10 and 20 (10 randomly generated problems per problem size). Six subjects were tested in the real TSP and five in virtual TSP (two authors of the study were tested in both). Some subjects were naïve and had no previous experience with solving TSP or with virtual reality systems. All subjects were tested with the same problems, but the order of the problems was different for different subjects. The subject's task was to collect the tennis balls in the order that produced as short a tour as possible. The subject was asked to return to the position of the starting ball in order to produce a closed tour. The same problems, involving the same city positions and distances were used in the virtual reality replication of the experiment. Considering the fact that the Cave has horizontal dimensions of roughly 3 by 3 meters, small movements of the subject around the simulated tennis court could be executed by walking around the Cave. Larger movements could be simulated by using a wand. The subject could also use the wand to rotate the scene around her. The subject produced the tour by collecting the tennis balls. As a result, the subject could not undo a move or start over, if he or she did not like the tour produced so far.

Results.

Tour errors for the real and virtual settings are shown in Fig. 6a and b. Consider first the TSP on a real tennis court. Except for one subject (MB), who missed a ball in a few problems with 10 and 20 cities, the performance of all subjects was quite consistent and similar to the performance when the problems are represented on a computer monitor (Pizlo et al., 2006). Large errors in the case of MB were caused by the fact that he started solving the TSP problems without looking around. As a result, when the subject was ready to finish a tour, and return to the starting city, he realized that there is one additional tennis ball (city) to collect. The subject was not allowed to "go back" and correct the tour because the visited "cities" were not there, anymore (the visited tennis balls have already been collected). As a result, the subject's tour was substantially longer than the optimal tour. This did not affect substantially the proportion of optimal solutions for this subject, but it did affect his average error.



a) Subjects' performance in reality

b) Subjects' performance in VR

c) MST pyramid model's performance with different parameter r

d) Average performance of subjects and the average performance of the MST pyramid model

(a) (b)

Figure 6. (a) Percentage error of subjects in real 2D TSP. (b) Percentage error of subjects in virtual 2D TSP. (c) Percentage error of the model for several degrees of local search. (d) average performance of subjects in real, and virtual TSP, as well as that of the model.

The average errors in the case of real TSP are also very close to the errors of solutions when the problems are presented on a computer monitor. These results imply that the errors in reconstructing the actual distances from perspective images do not affect the quality of the solution and, furthermore, that the subjects are able to integrate the visual information across the visual field. Apparently, accurate depth perception is not needed for solving TSP. Recall that the main feature of our computational model is hierarchical clustering. The model does not compute the overall length of the tour. Instead, the model generates, recursively, a tour of clusters. Apparently, (i) the clustering method used by humans is robust in the presence of reconstruction errors, and (ii) the top-down method of producing a tour do not strongly depend on the errors in evaluating distances.

It is also important to point out that performance of 5 out of 6 subjects in the case of real TSP problems was high despite that fact that they could not undo movements or start over. This is consistent with our observations from experiments where the TSP was solved on a computer monitor, namely that such corrections are really made. Once the subject decides how the problem should be solved, the subject executes the plan. The global aspects are decided first, and local decisions about the path are not likely to affect the global features of the path.

Results in the virtual reality replication of this experiment are similar, although there is more variability both within and across subjects. The average errors for individual subjects are shown in Figure 6b. The range of the errors is larger in VR than in real version of TSP. Specifically, the maximal average errors was 3% for 5 cities, 4% for 10 cities and 6% for 20 cities. These maximal errors are only slightly worse than those in the case of TSP on the real tennis court and TSP on a computer monitor. These results suggest that the VR environment is a reasonable representation of the real environment.

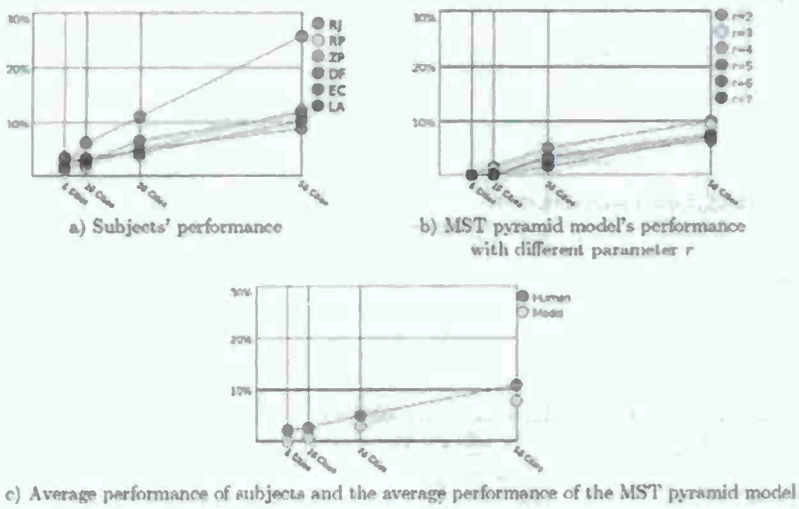
Next, we tested human and model's performance on 3D TSP, that is, TSP problems in which the cities are distributed in a volume. To the best of my knowledge, human subjects have never been tested with such problems. Results of this experiment have important implications for models of human performance. At this point, there are three such models. First, is the multiresolution pyramid, proposed by Pizlo and colleagues (Graham et al., 2000; Pizlo et al., 2006), second is the convex hull model by MacGregor et al. (2000) and third is the crossing avoidance model by van Rooij et al. (2003). Only the pyramid model can generalize to 3D space. The convex hull in 3D is not a tour, but a surface. So, the convex hull model cannot be applied to 3D problems. Crossing avoidance cannot be a useful strategy, either because a polygonal line connecting n points in 3D space is very unlikely to produce self-intersection. If any of these two models were a plausible explanation of how humans solve TSP problems, the tours produced by humans, in the case of 3D TSP, should be comparable to random tours.

3.2 Experiment on 3D TSP.

Only a virtual version of the TSP problems was used because it is difficult to place multiple points or objects in a real 3D space. Spheres with diameter of 15cm were generated within a cube 3x2x3 meters in the Cave VR system at Purdue. The number of cities (problem size n) was 6, 10, 20 and 50. There were 10 randomly generated TSP problems for each problem size. These problem sizes were used to allow a direct comparison with the results collected by our group for 2D TSP on a computer monitor (Pizlo et al., 2006). The viewing was binocular and the subjects were free to move. The subject used a 3D pointer to indicate the order in which the cities were visited. The subject could undo a move recursively. Three subjects were tested. All of them had extensive experience with virtual reality systems.

Results

Average solution errors of the six subjects tested on a 3D TSP are shown in Figure 7a. First, note that the three subjects produced consistent results. Next, note that these results appear only slightly worse than those for 2D TSP in a virtual environment. This contrasts with subjective reports of the subjects. Whereas 2D TSP appears natural and easy, 3D TSP appears very unnatural and difficult. The subjects reported that they had very little confidence in what they were doing in the case of 3D TSP and that they expected their tours to be no better than random tours. This was clearly not the case. For example, an average error of a randomly generated tour for 10-city TSP is 85%. In contrast, the average error of our subjects in the case of 10-city 3D TSP was less than 3%. The fact that 3D TSP *appears* unnatural is not surprising because it *is* unnatural. Human beings are used to solving 2D visual navigation tasks on surfaces, such as floors and ground. 3D visual navigation tasks have not been common, if present at all, during the evolution. Apparently, the visual (mental) mechanisms that are involved in solving 2D navigation tasks can be applied to 3D tasks, as well. This is not completely surprising to pilots, who are flying in 3D space and have to solve 3D tasks.



a) Subjects' performance

b) MST pyramid model's performance with different parameter r

c) Average performance of subjects and the average performance of the MST pyramid model

Figure 7. (a) Average error of six subjects in 3D TSP. (b) average error of the model. (c) average error of the subjects compared to the average error of the model.

What is the nature of the mental mechanisms involved? According to our computational model, human subjects solve TSP problems by means of: (i) hierarchical clustering, and (ii) top-down process of coarse-to-fine approximations to the solution tour. Our computational model has been designed for 2D TSP problems (Graham et al., 2000; Pizlo et al., 2006; Haxhimusa et al., 2007). Will the model generalize to 3D problems? Will it show a slight deterioration of performance, as compared to 2D TSP?

4. Graph pyramid model of TSP.

We took our recent model involving a graph pyramid and clustering based on the minimum spanning tree (Haxhimusa, et al., 2007). The main stages of the model, i.e., MST hierarchical clustering and the top-down sequence of tour approximations is shown in Figure 8.

This model had been tested on 2D Euclidean TSP and it provided a good fit to the results of the subjects. Because of the fact that this model uses graph representation of a TSP, it works the same way regardless whether the embedding is 2D or 3D. We ran this model on both types of problems and its averaged errors are shown on Figures 6 and 7. It can be seen that the model's errors in the case of 3D TSP are slightly larger than those in the case of 2D TSP. This result closely resembles the results of the subjects. The fact that the subjects' performance is systematically worse than the performance in the model can be attributed to difficulties in using the VR tools. Recall that the subjects' performance in real 2D TSP was higher than that in the case of virtual 2D TSP. The fact that human performance in 3D problems is only slightly worse than that in 2D problems provides further support to our pyramid models and it suggests that the convex hull and crossing avoidance models are not good models of the underlying mental mechanisms.

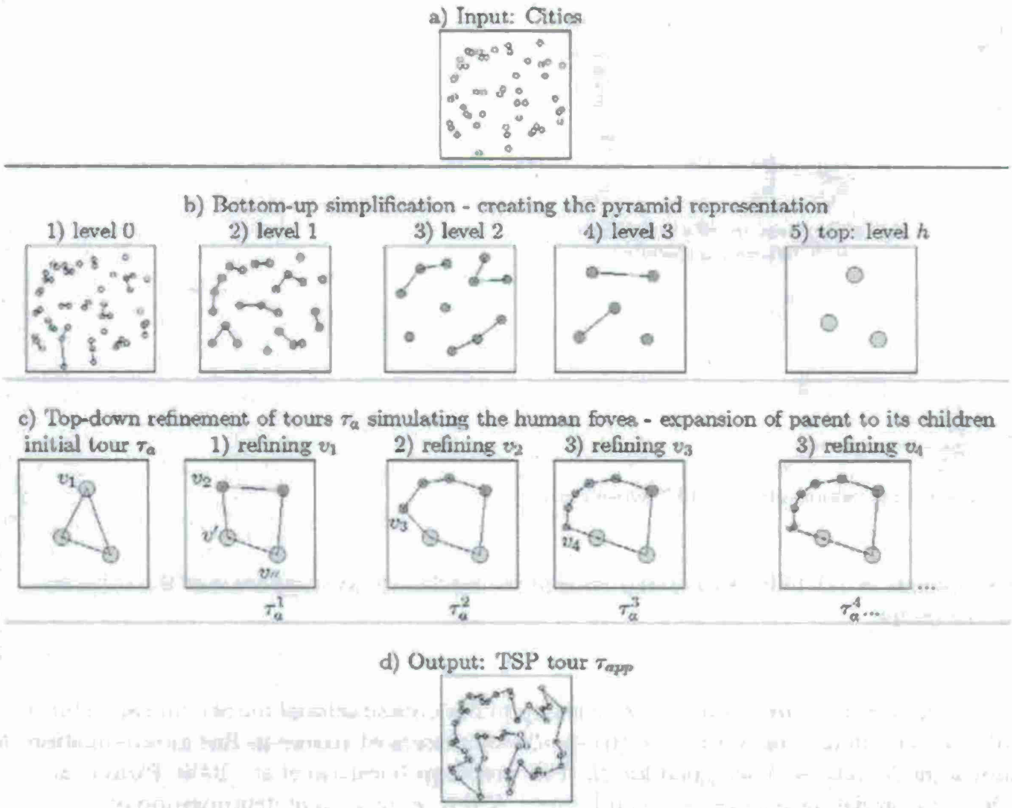


Figure 8. A 2D TSP problem is shown in (a). Hierarchical clustering is shown in (b). The coarse-to-fine sequence of tour approximations is illustrated in (c). The tour produced by the model is shown in (d).

5. Visual navigation and figure-ground organization

The TSP found somewhat unexpected applications in other areas of Cognitive Psychology, namely in visual navigation and visual figure-ground organization. Consider visual navigation, first. In everyday life, human beings are faced with the task of planning and executing the navigation between two points, as in the case of exiting a room, and planning a tour, as in the case of grocery shopping. In both tasks the human operator has to recover the 3D environment that includes multiple objects (obstacles). Once the map of the environment is ready, the task of planning the navigation is very similar to the shortest path problem and TSP, in the presence of obstacles.

The second application is intuitively less obvious. Consider the task of finding in a 2D image an occluding contour of a 3D object. This task is computationally difficult because the images always contain noise and occlusion. As a result, the contour is never a continuous curve. It is clear that the visual system must perform interpolation. All previous efforts relied on local interpolation methods, such as proximity and collinearity. These methods do not produce reliable results simply because local (greedy) methods do not capture the essential aspect of shape, which is its global nature. It seems that the TSP can do the job. This is illustrated in Figure 9. If a continuous, true contour of a 2D shape is sampled by a set of points, the TSP of these points is very likely to provide a good approximation to the true contour. This is true even if the sampling is quite sparse.

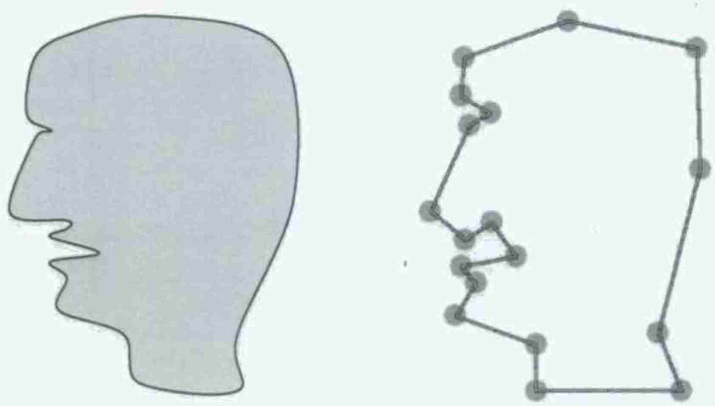


Figure 9. (a) An "ideal" 2D image of an object (human head). A "realistic" image is represented by the red points (cities). The green line is a TSP tour of the cities.

6. The effect of uncertainty on the quality of the TSP solution.

The problem can be stated as follows: how important is it to find the shortest tour in realistic cases of TSP problems, considering that the city positions are not known exactly? There has been a lot of work on improving optimal TSP algorithms, i.e., algorithms that find the shortest TSP tours. The main effort was devoted to formulating algorithms that run faster and faster. The time is a critical measure because an algorithm that guarantees finding the shortest tour has an exponential complexity. Our pyramid algorithm does not guarantee finding the shortest tour, but its complexity is linear. Our direct tests showed that a 1000 city problem is solved approximately by our algorithm in 40 sec (on average), whereas an optimal algorithm solves it, in more than 20 min. This difference could be important in some real-life applications. Note that when the city positions are not known exactly, then an "optimal" algorithm is not likely to find an optimal (shortest) tour. But then, why use an exponential-time algorithm to find an approximate solution when the approximate solution can be found by a polynomial-time algorithm?

We ran both simulation and psychophysical experiments using a design motivated by our experiments on 2D real TSP (see above). Assume that a problem solver is presented with a TSP problem on a slanted plane but the solver does not know about the slant. The problem solver will be solving the projected version, but its solution will be evaluated by using the original version. The comparison of the performance between the "optimal" algorithm (Concorde) and our pyramid model is shown in Figure 10. This graph shows the average and standard deviation of error on a 20 city problem, as a function of the slant of orthographic projection. The average errors produced by the two algorithms are very similar for slants up to 60 deg. The error bars suggest that on a number of problems, our approximating algorithm outperformed the "optimal" algorithm. Clearly, when the input data is not known exactly, there is no point in using an "optimal" algorithm which takes more time to run.

How realistic are these simulations? In other words, how likely is it that the errors in estimating distances will be comparable to errors produced by orthographic projection with a slant of 50 or 60 deg? It was pointed out above that when a problem solver is presented with a real 2D TSP under slant 80 deg, a 10 deg error in estimating the slant leads to 50% errors in judging distances in the direction orthogonal to the axis of rotation. This is equivalent to the error produced by an orthographic projection with slant 60 deg (cos(60deg)=0.5). This means that the results shown in Figure 7 represent the kind of situations that are likely to occur when a human or a robot are faced with a real 2D TSP.

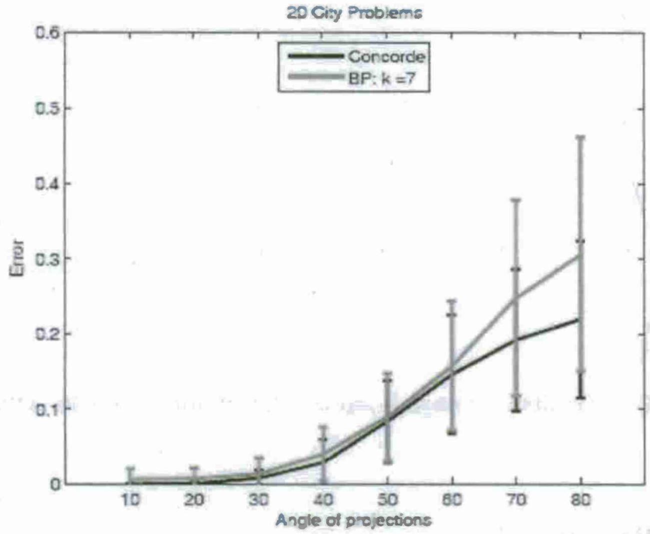


Figure 10. Mean and standard deviation of errors produced by an "optimal" algorithm (black) and by our pyramid algorithm (red). Both solved slanted TSP problems, but their performance was evaluated by using the original (not slanted) versions.

We tested two subjects (one naïve) on a subset of slanted TSP problems. Figure 11 shows the average errors for the two subjects and for the two algorithms (in this graph, performance of the algorithms on the same problems, is shown). Clearly, the subjects' errors are similar to the errors of the "optimal" algorithm. In fact, one subject (TS – the naïve subject) outperformed the optimal algorithm for slants 50 and 60 deg. Recall that the human mind uses a linear-time algorithm.

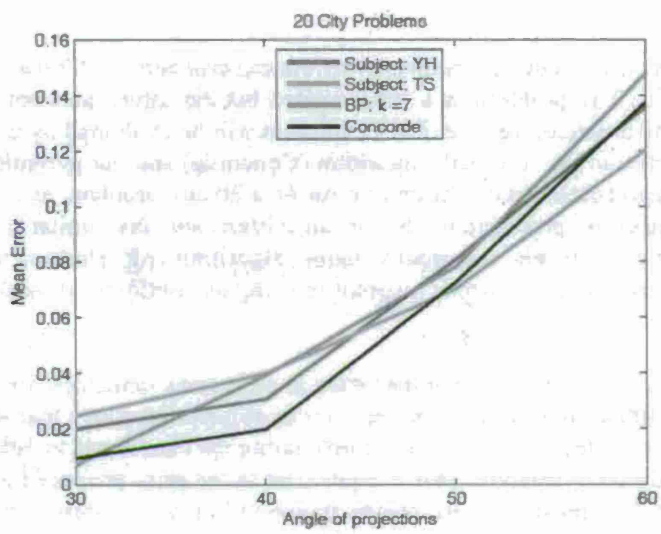


Figure 11. Mean errors produced by an "optimal" algorithm (black), our model (red) and by two subjects, for the range of slants between 30 and 60 deg.

7. Limited memory TSP algorithm.

Our pyramid model solves the TSP problem by emulating a non-uniform distribution of visual acuity in the human visual system. As a result, the model does not have access to the entire search space at the same time. It has access only to a coarse representation of large parts of the original problem. A high resolution representation is limited to only a very small part of the search space. This limitation is compensated by the simulation of eye movements and attention. Despite this "software" limitations of the model, the algorithm actually represented the entire problem in its memory. This is surely different in human agents. When they solve a 50- or 100-city TSP, it is unlikely that the visual system stores the entire problem in its visual memory. It is not needed because the problem is always in front of the observer, so if the observer wants to analyze a given part of the problem, he can actually look at it. There is no need to memorize it. The same is true with other complex problems such as chess playing. A chess player keeps in his memory only a small fraction of the game.

In order to emulate this ability, the model keeps changing its memory content depending on the current stage of the problem solution. In other words, a video camera is simulated and this camera provides input data about the problem. It can provide a low resolution representation of the entire problem or high resolution representation of a fraction of the problem. In a sense, the problem is represented as a discrete image whose resolution dynamically changes. This leads to new challenges in solving such problems as hierarchical clustering. The clustering is performed for a digital image that represents intensity distribution, rather than discrete points. An illustration of this process is shown in Figure 12.

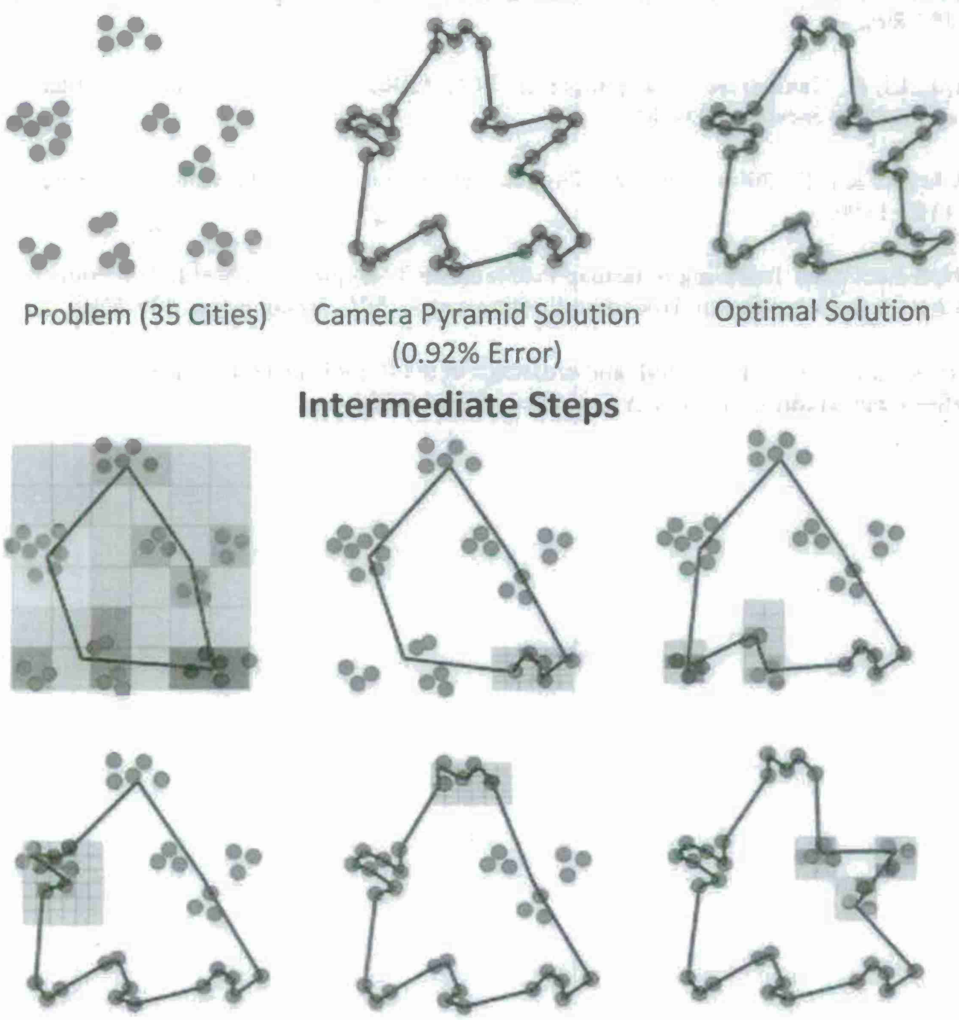


Figure 12. Top row: the problem, its solution by the limited memory model and the optimal solution. The next two rows illustrate the solution process. The process begins with 5 clusters coded by different colors. One of the clusters (bottom right) is chosen as the starting cluster. Then, the solution proceeds clockwise.

The fact that the limited memory pyramid model can produce near-optimal solutions suggests that this kind of model can, in principle, be applied to other cases characterized by very large search spaces. It is an open question as to how many levels such a pyramid may have. Is the number of levels limited? This could be the case with human problem solvers. Note that a human problem solver is not likely to "handle" a TSP problem with 10,000. cities, although, our pyramid model can do that. Such large problems as 10,000-city TSP would require a pyramid with large height. Assume that a bisection pyramid is used, as in Pizlo et al. (2006). The height of such a pyramid for 10,000-city problem is expected to be $\log_2 10000 \approx 14$. This contrasts with a 100-city problem for which the height of the pyramid is about 7. Perhaps the span of human short term memory represents a height of the perceptual pyramid.

Refrences

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

Haxhimusa, Y., Kropatsch, W.G., Pizlo, Z., Ion, A. & Lehrbaum, A. (2007) Approximating TSP solution by MST based graph pyramid. Proceedings of the IAPR Workshop on Graph-Based Representations in Pattern Recognition, *Lecture Notes in Computer Science*, Escolano F. & Vento M. (Eds.), vol. 4538, (pp. 195-306). Berlin: Springer.

Haxhimusa, Y., Kropatsch, W.G., Pizlo, Z., & Ion, A. (2009) Approximate graph pyramid solution of the E-TSP. *Journal of Image and Vision Computing* **27**, 887-896.

Pizlo, Z., Stefanov, E., Saalweachter, J., Li, Z., Haxhimusa, Y. & Kropatsch, W.G. (2006) Traveling Salesman Problem: a Foveating Pyramid Model. *Journal of Problem Solving1*, 83-101.

MacGregor, J.N., Ormerod, T.C. & Chronicle, E.P. (2000) A model of human performance on the Traveling Salesperson Problem. Memory & Cognition, 28, 1183-1190.

Saalweachter, J. & Pizlo, Z. (2008) Non-Euclidean Traveling Salesman Problem. In: T. Kugler, J.C. Smith, Y-J. Sun, T. Connolly, Decision Modeling and Behavior in Complex and Uncertain Environments. NY: Springer (pp. 339-358).

Van Rooij, I., Stege, U. & Schactman, A. (2003) The convex hull and crossings in the Euclidean Traveling Salesperson Problem: implications for human performance studies. Memory & Cognition, 31, 215-220.