A NEW HYBRID HEURISTIC FOR LARGE GEOMETRIC TRAVELING SALESMAN PROBLEMS BASED ON THE DELAUNAY TRIANGULATION

Natalio Krasnogor [1]
CeTAD, Universidad Nacional de La Plata,
C.C. 75, 1900 La Plata, ARGENTINA

Pablo Moscato [2]
CeTAD, Universidad Nacional de La Plata,
C.C. 75, 1900 La Plata, ARGENTINA
email: mos@ada.info.unlp.edu.ar

Michael G. Norman
Makespan Limited, 8 Gayfield Square,
Edinburgh, EH1 3NT, Scotland, United Kingdom

ABSTRACT: In this paper we present a new hybrid heuristic for large geometric Traveling Salesman Problems. The method is based on the use of the *Delaunay Triangulation* of a set of cities. Many heuristics for the TSP are generally composed of two phases: a constructive heuristic followed by an iterative improvement step until a local minimum is reached. We think of this heuristic as a hybrid since, after a city has been introduced, a period of local improvement takes us to another local minimum before a new city is added. We provide computational results using many TSPLIB geometric instances.

KEYWORDS: Traveling Salesman Problem, Heuristics, Delaunay Triangulation.

RESUMO: Neste artigo, apresentamos uma nova heurística híbrida para os problemas geométrico do Caixeiro Viajante de grande porte. O método é baseado no uso de "Delaunay Triangulation" de um conjunto de cidades. Muitas heurísticas para o problema do Caixeiro Viajante são geralmente compostas por duas fases: uma heurística de contrução e um passo de melhoria, até que um ótimo local seja alcançado. Nós pensamos nesta heurística como híbrida desde que, depois que uma cidade é introduzida, um período de melhoria local nos leva a outro mínimo local antes que uma nova cidade tenha sido adicionada. Fornecemos resultados computacionais usando instâncias do TSPLIB.

PALABRAS-CHAVE: Problema do Caixeiro Viajante, Heurísticas, "Delaunay Triangulation".

Voronoi Diagram and Delaunay Triangulation

The *Voronoi diagram* solves the *"loci of proximity"* problem: *"Given a set S of N points in the plane, for each point p in S; which is the locus of points (x, y) in the plane that are closer to p than to any other point of S ?"*. Given two points p and q, the set of points closer to p than to q is just the half-plane containing p that is defined by the perpendicular bisector of $\overline{pq}$. Let us denote this half-plane by $H(p, q)$. The *Voronoi polygon associated to a point p* (denoted $V_R(p)$) is a convex polygonal region having no more than $N-1$ sides since it is the intersection of $N-1$ bisecting half-planes defined by p and the other $N-1$ points in S. Then

$$V_R(p) = \bigcap_{p,q \in S, p \neq q} H(p, q) \tag{1}$$

Given the *Voronoi diagram* of S there is a graph which is its *straight-line dual*. It is called the *Delaunay Triangulation* (DT) denoting it as (S, D_T). It is a graph that has S as the vertex set and a set of (undirected) edges D_T, where $D_T = \{\{p, q\} \mid V_R(p) \bigcap V_R(q) \neq \emptyset\}$. Alternatively, the *Delaunay Triangulation* of a set of vertices in the plane consists of all line segments $\overline{pq}$ such that there exists a circle passing through p and q containing no other vertices. Equivalently, $\overline{pq}$ is a Delaunay edge if and only if the Voronoi regions for p and q are adjacent. We recall here that the Voronoi region of a vertex p is the polygon with the property that all points inside it are closer to p than to any other vertex. In this sense, the Delaunay edges encompass some notion of *"neighborhood"*. We have implicitly assumed that no two vertices are equal, that no three vertices lie on a straight line, and

[1] also at LIFIA, Dept. of Informatics, UNLP
[2] Prof. de Apoyo a la Inv., C.I.C.P.B.A.

that no four vertices lie on a circle. If those nondegeneracy conditions are not satisfied, the standard accepted procedure is to perturb the points very slightly so that the assumptions do hold.

The *Delaunay graph*, which we will note as (S, D_G), is created from (S, D_T) by excluding those edges $\{p, q\} \in D_G$ for which $| V_R(p) \bigcap V_R(q) | = 1$ (that is the Voronoi polygons of the two vertices meet at a single point). In contrast to (S, D_T), (S, D_G) is no longer a triangulation but it is guaranteed to be planar, implying that $|D_G| = O(N)$.

OCIDT: A hybrid heuristic

More than a decade ago, it has been shown that the optimal tour of a planar point set is not necessary a subgraph of the DT [1], thus finally settling an open conjecture. In spite of this fact, we have conducted some preliminary tests using the TSPLIB instances [3] to empirically estimate how many edges which belong to the optimal tour are not present in the DT. Unfortunately, there is a small set of TSPLIB instances which have been solved to optimality and for which a file with the optimum tour has also been provided in the standard, public domain, release. However, some preliminary results show evidence that this percentage is very low. This indicates that the DT has, in general, some good information about the *"candidate set"* of possible edges to be considered as being part of the optimum tour. The use of candidate sets is a standard technique to improve the efficiency of iterative improvement heuristics and metaheuristics based on Tabu Search (see Ref. [2]). In the context of the DT they have been applied in [3].

Our interest in the DT also derived from the consideration of another heuristic, FAMCH, in [4]. FAMCH is a variant of a the heuristic studied by D.S. Johnson in Ref. [5].

> **Farthest Addition from Minimal Convex Hull (FAMCH)**: Start from the (possibly incomplete) tour comprising the minimal convex hull (that is the convex hull from which have been removed all vertices to which adjacent edges are at an angle of 180 degrees). Repeatedly choose the non-tour city with the maximal distance to its nearest neighbor amongst the tour cities, breaking ties randomly, and insert it between the two consecutive cities in the subtour for which such an insertion causes the minimum increase in total tour length, breaking ties randomly.

In Ref. [4] it has been applied to four "fractal" TSP instances of different box-counting dimensions. We have proved that this heuristic outputs the optimal solution of KochTour, the instance with the lowest fractal dimension in that study. It failed to return the optimum tour in two instances with box-counting dimension of two and it also failed in DavidTour, an instance with empty regions at all scales.

It was this question of the appropiate insertion order that led us to start considering different variants of the selection procedure given above: *"Repeatedly choose the non-tour city with the maximal distance to its nearest neighbor amongst the tour cities, breaking ties randomly."* The DT was a good candidate for a proper definition of neighborhood of a point which belongs to a set of points. Early on, we have noticed that it was better to introduce first those cities which have low degrees in the DT. We thought that this was quite reasonable and we soon established this procedure as the first selection criteria for insertion.

There is an interesting thing to point out about this strategy. It is known that we can associate a polyhedral surface to every triangulation, in particular the DT, by requesting that all links be unit length. It is clear that that surface, composed of equilateral triangles, can not be flat since the only case which is flat is a city that has 6 neighbors in the DT. Five neighbors would have a positive curvature, while seven would form a saddle which has negative curvature [6]. This said, we have the degree of a city in the DT as the first selection criterion to rank the cities to be inserted and it may be reinterpreted or better "refined", in future studies, with the concept of curvature as well as others discussed in Ref. [6].

[3]TSPLIB is a database compiled and maintained by Gerhard Reinelt of the "Institut für Angewandte Mathematik, Universität Heidelberg", E-Mail `Gerhard.Reinelt@IWR.Uni-Heidelberg.DE`. TSPLIB contains many TSP instances solved to optimality.

We will need another criteria to derandomize the selection since, obviously, there are many cities with the same number of neighbors in the DT. For the moment, we will call this criteria as $H\#$ and make explicit the basic OCIDT structure (the name OCIDT is simply derived from *"One-City-Insertion from DT"*).

Basic OCIDT

1. Create the Delaunay Triangulation of the set of points S.
2. Repeat until all the points have been inserted.
   (a) Find the city not yet in the tour that has the lowest degree in the DT. Break ties regarding $H\#$.
   (b) Insert the selected city between the two consecutive cities in the subtour for which such an insertion causes the minimum increase in total tour length, breaking ties randomly.
   (c) Start an iterative improvement *one-city-insertion* step.

We remark the fact that the iterative improvement step is realized each time a city is added, thus drastically reducing the insertions to check. It is interesting to see how the heuristic evolves; sometimes a *"cascade of events"* changes dramatically the overall shape of a partial tour. The heuristic has a low complexity since the DT can be calculated in $O(NlogN)$ time. We have selected to use the *one-city-insertion move* (that is to check for the insertion of one city between other two) due to its simplicity and its low order complexity. Other type of moves, like the 2-change of 2-Opt, would also be applied but are not part of this study. It may be the case that a more powerful iterative improvement technique may improve the basic OCIDT scheme, but we have chosen the OCI to avoid masking the effects of the insertion order.

In the TSP literature, constructive and iterative improvement heuristics are generally applied in *"tandem"*, since the latter is usually applied to a tour on the complete set of cities. This is a clear difference with other methods. In our case, the working hypothesis of our hybrid method is that we may find a way to exploit the correlation of local minima of two partial tours which only differ in one city. The relevance of the correlation of local minima for some metaheuristics has been discussed in Ref. [2]. F. Glover also refers to it as the *"Proximate Optimality Principle"*. Naturally, the heuristic then faces us with the problem of selecting a proper insertion order.

In this paper we will report computational results with two different selection criteria (H1 and H2) thus defining two heuristics. In both cases we compute for each city the sum of the lengths of all edges in DT that have that city as an endpoint. Using this sum, H1 ranks the cities in decreasing order while H2 does it in increasing order. When many cities can be selected, due to the fact that they have the same number of neighbors, H1 inserts them in decreasing order while H2 does it in the opposite order. We have done this to check for the consistency of the sum of edge lengths as a useful measure for derandomization of the selection criteria based on the degree of cities in the DT.

Computational Results

In the following tables, all instances have a name contaning their number of cities, i.e. `att532` is the TSP instance of 532 cities known as `att532.tsp` in TSPLIB. The column "Bounds" gives known bounds for the optimum tour length when known (coming from a variety of sources, see the Acknowledgments in Ref. [3]). The next column ("Gap") indicates the gap between lower L and upper bound U and is given in percent (computed as $100(U-L)/L$). Of course, when only one number is provided that number is indicating the length of the optimal tour.

For comparison, we think it is illustrative to include G. Reinelt's results since he has made use of the DT in a different way. He uses the DT and the DG to define a subgraph of the complete graph among all cities and he developed a variant of the classic Nearest Neighbor heuristic. He preempts on the insertion of nodes as soon as they become "isolated", that is the case of a node not yet inserted which has 2 or 3 free edges in the subgraph.

We have chosen to display our results by separating them in two groups. In Table 1 we are presenting our results in most of the instances used by Reinelt [3] for a better comparison. In addition, we have conducted a similar study for many other geometric instances from TSPLIB. These results are presented in Table 2. The results of Table 1 are those which belong to the third variant of the heuristics presented by Reinelt in Ref. [3] which we reproduce for comparison. He indicates that one as the method of choice since the average quality of 2-Opt and Lin-Kernighan tours is 7.31% and 3.59%, while the average value for NN is 18.69 %. Another heuristic which uses the degree limit of three leads to sligtly inferior results of 7.83% (2-Opt), 4.05% (L–K) and, 20.73 % for NN. For comparison we have included the data of Reinelt's best results which were presented in Table V of Ref. [3]. The columns H1 and H2 contain the length of the tour found by the heuristic and the Gap using the lower bound or the optimum value. From Table 1 we can see that the method compares relatively well with the 2-Opt technique as implemented by Reinelt who has used as starting tours those which where the output of the NN method. These tours, it was recognized by Reinelt, *"are usually locally not bad and contain only a few severe global errors that can be corrected easily (and quickly)."* We must remark that our tours are neither 2-Opt nor optimal under the L-K procedure thus we still have a margin to improve them using the same techniques. Another thing that can be induced from a careful examination of both tables is that we expected H1 to be much better than H2 and the computational results would indicate that the gap is not so big. This somewhat debates our selection of the sum of edge lengths as a proper selection criteria and leaves the issue open for further research on the use of other features from the Delaunay Triangulation.

Table 1

Name	Bounds	Gap	NN	2–Opt	L–K	H1	H2
d198	**15780**	–	17.00	3.86	7.52	16357 (3.66)	16462 (4.32)
pcb442	**50778**	–	11.99	5.26	2.50	53590 (5.54)	54680 (7.68)
u574	**36905**	–	18.16	8.00	4.26	39341 (6.66)	39170 (6.13)
p654	**34643**	–	32.94	3.40	1.97	36287 (4.75)	36805 (6.24)
rat783	**8806**	–	19.50	6.72	2.96	9497 (7.85)	9462 (7.44)
pcb1173	**56892**	–	15.21	6.22	2.73	61513 (8.12)	61725 (8.5)
d1291	[50606,50864]	0.51	14.95	6.43	1.96	56227 (11.11)	55987 (10.63)
u1432	**152970**	–	17.71	8.33	3.13	165218 (8.01)	163337 (6.77)
fl1577	[22134,22249]	0.52	15.57	9.12	2.94	24321 (9.88)	24422 (10.34)
d1655	**62128**	–	18.50	7.08	3.86	67959 (9.39)	68696 (10.57)
d2103	[79687,80259]	0.72	9.30	3.91	3.02	92629 (16.24)	93317 (17.10)
pr2392	**378032**	–	20.58	7.70	4.19	408622 (8.09)	406252 (7.47)
pcb3038	[137617,137694]	0.06	19.95	8.15	3.27	148964 (8.25)	148540 (7.94)

Acknowledgment - Part of this work was done while M.N. was in Argentina and his visit was funded under an agreement between the Royal Society (U.K) and CONICET (Argentina). P.M. wants to thank R.E. Berretta for her help during the preparation of this paper.

REFERENCES

[1] V. Kantabutra. Traveling salesman cycles are not always subgraphs of Voronoi duals. *Inform. Process. Lett.*, 16:11–12, 1983.

[2] P. Moscato. An introduction to population approaches for optimization and hierarchical objective functions: The role of tabu search. *Annals of Operations Research*, 41(1-4):85–121, 1993.

[3] G. Reinelt. Fast heuristics for large geometric traveling salesman problems. *ORSA Journal on Computing*, 4(2):206–217, 1992.

[4] P. Moscato and M.G. Norman. An analysis of the performance of traveling salesman heuristics on infinite-size fractal instances in the euclidean plane. Technical report, CeTAD - Universidad Nacional de La Plata, 1994. submitted - Oct. 1994.

[5] D.S. Johnson. Local optimization and the traveling salesman problem. In *Proceedings of the 17th Colloquium on Automata, Languages and Programming*, volume 443, pages 446–461, Berlin, 1990. Springer-Verlag. Lecture Notes in Computer Science.

[6] M.O. Magnasco. Correlations in cellular patterns. *Philosophical Magazine B*, 69(3):397–429, 1994.

Table 2

Name	Optimum	H1	H2
eil76	**538**	556 (3.34)	567 (5.39)
pr76	**108159**	112342 (3.87)	112143 (3.68)
rat99	**1211**	1293 (6.77)	1293 (6.77)
kroa100	**21282**	23458 (10.22)	21391 (0.51)
krob100	**22141**	22576 (1.96)	22614 (2.13)
kroc100	**20749**	21675 (4.46)	21328 (2.79)
krod100	**21294**	21826 (2.50)	22106 (3.81)
kroe100	**22068**	22745 (3.07)	22509 (2.00)
rd100	**7910**	8240 (4.17)	8256 (4.37)
eil101	**629**	655 (4.13)	653 (3.82)
lin105	**14379**	15142 (5.31)	15628 (8.69)
pr107	**44303**	44577 (0.62)	44577 (0.62)
pr124	**59030**	60091 (1.80)	59458 (0.73)
bier127	**118282**	125663 (6.24)	122423 (3.86)
pr136	**96772**	102257 (5.67)	99262 (2.57)
pr144	**58537**	59576 (1.77)	60422 (3.22)
kroa150	**26524**	27355 (3.13)	27971 (5.46)
krob150	**26130**	26481 (1.34)	26832 (2.69)
pr152	**73682**	75018 (1.18)	75359 (2.28)
u159	**42080**	44067 (4.72)	44129 (4.86)
rat195	**2323**	2485 (6.97)	2534 (9.08)
d198	**15780**	16357 (3.66)	16462 (4.32)
kroa200	**29368**	30706 (4.56)	31379 (6.85)
krob200	**29437**	30771 (4.53)	30666 (4.17)
ts225	**126643**	138449 (9.32)	138532 (9.39)
pr226	**80369**	81724 (1.68)	81704 (1.66)
gil262	**2378**	2547 (7.10)	2516 (5.80)
pr264	**49135**	52083 (6.00)	52763 (7.38)
pr299	**48191**	50672 (5.15)	51209 (6.26)
lin318	**42029**	44665 (6.27)	43625 (3.80)
fl417	**11861**	12281 (3.54)	12468 (5.11)
pr439	**107217**	112453 (4.88)	113490 (5.85)
d493	**35002**	37129 (6.08)	36641 (4.68)
att532	**27686**	29097 (5.10)	29386 (6.14)
u574	**36905**	39341 (6.60)	39170 (6.13)
rat575	**6773**	7272 (7.37)	7237 (6.85)
d657	**48912**	51303 (4.89)	51581 (5.45)
u724	**41910**	44626 (6.48)	44573 (6.35)
rat783	**8806**	9497 (7.85)	9462 (7.44)
nrw1379	**56638**	59666 (5.34)	60450 (6.73)
fl1400	n.p.	20805	20721
u1432	**152970**	165218 (8.00)	163337 (6.77)
d1655	**62128**	67959 (9.39)	68696 (10.57)
fl3795	n.p.	32266	31519