Longest path estimation from inherently fuzzy data acquired with GPS using genetic algorithms.

Adolfo Otero, José Otero, Luciano Sánchez and José R. Villar
Computer Science Department
University of Oviedo
Gijón, Spain 33203
Telephone: (+34) 985182597
Fax: (+34) 985181986

***Abstract*— Measuring the length of a path that a taxi must fare for is not an obvious task. When driving lower than certain threshold the fare is time dependent, but at higher speeds the length of the path is measured, and the fare depends on such measure. When passing an indoor MOT test, the taximeter is calibrated simulating a cab run, while the taxi is placed on a device equipped with four rotating steel cylinders in touch with the drive wheels. This indoor measure might be inaccurate, as information given by the cylinders is affected by tires inflating pressure, and only straight trajectories are tested. Moreover, modern vehicles with driving aids such as ABS, ESP or TCS might have their electronics damaged in the test, since two wheels are spinning while the others are not.**

To overcome these problems, we have designed a small, portable GPS sensor that periodically logs the coordinates of the vehicle and computes the length of a discretionary circuit. We will show that all the legal issues with the tolerance of such a procedure (GPS data are inherently imprecise) can be overcome if genetic and fuzzy techniques are used to preprocess and analyze the raw data.

I. Introduction

One of the tasks to be performed in the Spanish VTSS is the test and control of the taximeters in the taxicabs. This supervision must be performed every year because the taxicabs' fares are revised and published by the authorities every year. The process a taxicab owner must follow includes driving the taxicab to a specialized garage to change the fares in the taximeter. When the fares are changed, a MOT test must be done. In this MOT test, the tester engineer verifies if both the distance traveled and the waiting time fares lie between the limits imposed. The verification of the fares can be done in two ways. The simplest way consists in doing a cab run in a previously measured circuit; in this case the MOT test engineer manually calculates the resulting fare, but the waiting fare and the traveled fare cannot be done at the same time, because the changing from one fare to the other depends on the speed of the taxicab. The second approach is to use a machine capable of the recovering of the speed of the cab to select the waiting fare or the traveled fare and to compute the time elapsed and the distance. In 1990 we developed a device to accomplish this task. The machine was made of four rotating steel cylinders, one of them connected to an encoder. The pulses from this transducer were fed to a computer where the appropriate software computed the fare

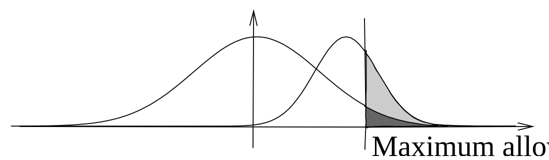

Fig. 1. If the owners of the taxis calibrated their taximeters in good faith, the density of the errors in the measures of taximeters should be centered in 0. Field measures show that the density is centered near 9% (the legal cut point is 10%). A small deviation in the tolerance of our measure, which would be unnoticed under theoretical circumstances (dark gray area,) will cause a high percentage of rejections (light gray area).

to apply (as well as the legal limits of the fare). To test a cab taximeter, the drive wheels of the cab were put between the cylinders and a cab run was simulated. This system is still in use but fails with modern cabs where electronic driven aids are present. Active safety systems nowadays present in cars like ABS, ESP, TCS are connected to electronic control units to process different signals. These signals include data, among others, of the speed of the four wheels of the car. If some difference is detected, the electronic control unit tries to help, sending messages to solve that, possible dangerous, situation. So if the two rear wheels are blocked and the two front wheels are rotating during a long time, the electronic control unit may be damaged. In this situation, a new method of testing taximeters must be developed. This system should be designed taking into account that it is not desirable to block one MOT test engineer when testing a taximeter. We have decided to use GPS technology to track the position of a vehicle in an actual road, and process this information on-line. Moreover, the taxi driver can be sent alone to cover a distance, and no personal of MOT agency is needed, making the process cheaper.

There are some drawbacks, though. GPS generates imprecise data, and the degree of imprecision of every sample is different. The differences in tolerances must be taken into account in the algorithm that analyzes the data. The significance of this step is crucial for our system to compute the upper and lower bounds of the length of the trajectory, which must be provided in the case that a a taximeter is rejected. The legal margin of error of a taximeter in Spain is 10%. We can not reject a taxi with a deviation of 7% if we can not warrant a tolerance lower than 3%, say. This could seem a minor problem, and it would be, if the density of the errors in the taxis resembled the left Gaussian in Figure 1. Unfortunately, our study revealed that the calibration of taximeters is far from unbiased. Small changes in the tolerance produce important changes in the number of rejections. Therefore, it is needed a procedure to determine the bounds of the measure with high accuracy and it is also needed that all the tolerance errors benefit the owner of the taxi. In other words, we need to compute the lowest upper bound (LUB) of the trajectories compatible with the (imprecise) GPS measures.

In this paper we will explain a new method for estimating the LUB of the trajectory from imprecise data. Through a multiobjective genetic algorithm, the measures are filtered to obtain the smallest set of samples that define a multi polygonal covering the input data. The LUB of the path is found by means of a deterministic algorithm that processes this multi polygonal. The structure of this work follows: In next section, how GPS measures are obtained is detailed. Then, a description of the proposal is done in section III. The genetic algorithm is detailed in section III-A, while the deterministic algorithm for estimating the maximum length is detailed in section III-B. In section IV experiment and results are shown. Finally, conclusions and future work are presented.

II. GPS-BASED MEASURES ARE FUZZY DATA

The term Global Positioning System (GPS) refers to a set of devices (satellites and receiver) working together to get a fix (the position) of the receiver. The receiver can get some signals from the satellites and compute a set of measures: longitude, latitude, altitude, number of satellites in use, time, etc. Each signal received from a satellite contains information about the time that the signal lasts from the satellite to the receiver.

So it can be thought that using signals from four satellites (three for geographical coordinates and one for time correction) could be enough to get a fix. One fix computed with that information, however, is pretty inaccurate: there are some errors in GPS technology that make necessary to get signals from more than four satellites. Some of the sources of these errors are: perturbations of the satellites' signals when crossing the atmosphere, satellite ephemerids deviation, satellites' clock errors, receiver errors and multipath (signals are not received directly from the satellite).

The higher the number of satellites, the better the accuracy. But even with a high number of satellites in use (12 to 16) the geometry or constellation of the satellites must be taken into account to estimate the fix accuracy. This is done using DOP (Dilution of Precision), a measure of the probability of the effects of the constellation on the fix accuracy; a higher value of DOP indicates a weaker geometry of satellites. DOP has four components: PDOP (3D or spherical DOP), HDOP (latitude and longitude DOP), VDOP (vertical DOP) ant TDOP (time DOP).

In the case of GPS longitude and latitude accuracy, the HDOP value must be taken into account.

When using consumer-grade receivers, it is very common to obtain accuracies like 3 meter CEP (50%) and 7 meters (90%), where CEP means Circular Error Probable or median error.

A. HDOP and GPS horizontal precision errors

As stated before, not only the number of satellites but also their relative positions has an impact on GPS accuracy. Here we explain how to estimate the impact on horizontal position. The term HDOP of DOP stands for Horizontal Dilution Of Precision and is close related to the horizontal precision of GPS receivers, specifically to the CEP magnitude.
The RMS error of a set of fixes under a common HDOP can be approximated by equation 1, where A and B are constants hardware dependent, empirically obtained [22].

$$RMS\_Error(HDOP) = \sqrt{(A \cdot HDOP)^2 + B^2} \tag{1}$$

HDOP is related with the number of available satellites through equation 2, where C and D are hardware dependent again.

$$RMS\_HDOP = \frac{C}{(\text{number of satellites})^2} + D \tag{2}$$

The distribution of the position measured by a GPS device follows a Rayleigh error probability conditioned to a given HDOP, as expressed in equation 3, where the different magnitudes have been defined in previous paragraphs.

$$P(Err \leq CEP|HDOP) = 1 - e^{\left(\frac{CEP}{RMS\_Error(HDOP)}\right)^2} \tag{3}$$

From equation 1 and 3 and solving for CEP, follows that CEP can be computed by equation 4, where for a given probability and a HDOP the corresponding CEP is obtained. An analogous procedure with equation 2 leads to an expression where the number of satellites appears instead of HDOP [22].

$$CEP = \left(-((A \cdot HDOP)^2 + B^2) \cdot \ln(1 - P(Err \leq CEP|HDOP))\right)^{0.5} \tag{4}$$

B. Fuzzy interpretation of GPS-values

Under the imprecise probabilities framework, it makes sense to understand a fuzzy set as a set of tolerances, each one of them is assigned a confidence degree, being the lower degree the narrower tolerance [12]. In particular, it has stated that, given an incomplete set of confidence intervals for a random variable, we can build a fuzzy random variable, whose α-cuts are confidence intervals with degree $1 - \alpha$ [5], that contains all the information we know about the unknown random variable. In our case, the GPS sensor provides two confidence intervals at 50% and 90% (the mentioned circle of radius CEP,) and therefore the fuzzy representation of GPS coordinates is immediate.

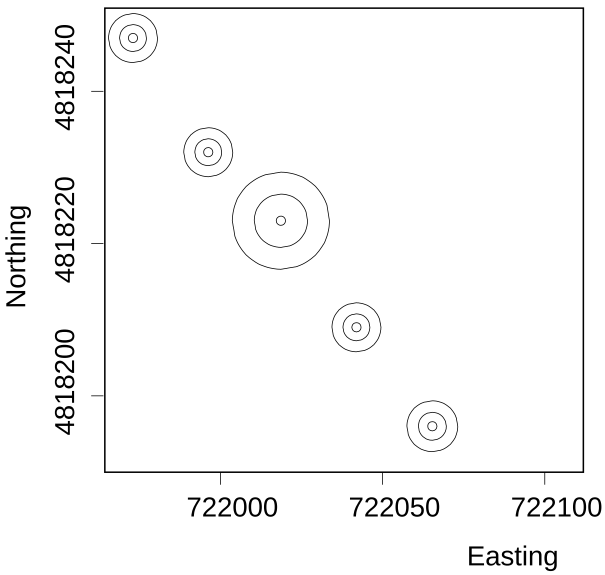

Fig. 2. Actual data in table I, the circles represent CEP in meters, the units of x and y axes.

Easting	Northing	CEP 0.90	CEP 0.50
722064.9	4818196	7.834569	4.298530
722041.5	4818209	7.547423	4.140984
722018.2	4818223	14.964010	8.210182
721995.8	4818232	7.547423	4.140984
721972.6	4818247	7.547423	4.140984

TABLE I
REAL DATA PLOTTED IN FIGURE 2, FIRST TWO COLUMNS ARE UTM EASTING/NORTHING COORDINATES, LAS TWO COLUMNS ARE CEP FOR 0.90 AND 0.50 PROBABILITY

III. DETERMINING THE LENGTH OF TRAJECTORIES USING FUZZY DATA

GPS data is recorded at regular time intervals. Each sample is a fuzzy set, as mentioned, whose cuts are circles. In turn, every circle is a confidence interval for the coordinates of the taxicab at that moment. It is remarked that taking the centers of these circles is not a valid estimation. We need to compute the LUB of the paths whose extremes are contained in the circles, and this length will always be higher than the value obtained from the centers.

The answer to the problem is not easy, though. If we try to compute the maximum length of all compatible piecewise linear paths that are contained in the circles it is obvious that, the shorter the sampling period, the longer the estimation. This is not correct, and we wish the estimation of the length not to be too influenced by the sampling period [16]. We have decided to preprocess the fuzzy data and remove all redundant information with the help of a genetic algorithm, as we will show in the section that follows.

When using crisp data, the geometric problem of simplifying polygonal lines has been studied in [10], [14]. The most similar approach to ours, up to our best knowledge, uses fuzzy data from a geographical database [1] for reconstruction of 3D images by means of B-splines [3]. By extending B-splines with fuzzy coefficients, and training those fuzzy numbers minimizing the fuzzy data that are not covered, the resultant fuzzy B-spline interpolates between its limits. A fuzzy point is said to be covered by the fuzzy B-spline if the fuzzy set induced by the latter completely contains the former.

A. Multiobjective fuzzy fitness genetic algorithm for filtering the fuzzy input data

The input of the genetic preprocessing is a vector of fuzzy points. The output is the minimum set of fuzzy input data that defines a fuzzy trajectory containing as many points as possible. Using those fuzzy points, and for each α-cut, a distance value is computed by means of a deterministic algorithm, which will be detailed later.

Every candidate solution is evaluated as follows: we first build a polygonal chain for each α-cut of the selected data, using the tangent surfaces to the selected fuzzy data set. [1] We wish that this chain contains as many data as possible, while having the minimum area.

Both objectives are fuzzy numbers and define a multicriteria problem [4] which we will solve by means of the NSGA-II algorithm [6], [7]. Further details of this algorithm follow.

1) Codification of each individual: Since we wish to obtain a subset of the input fuzzy data, the representation used for each individual is a vector of integers indexing the input vector, determining the fuzzy data used to define the polygonal chain. This representation admits each individual to have a different number of final fuzzy data. To generate an individual, a probability p value is given, and for each fuzzy point in the vector of input fuzzy data, it is included in the hypothesis with probability less of equal than p. The origin and the end of the ride must be always included.

2) Genetic operators: The definitions of the crossover and mutation must reduce the number of vertexes in the population, and therefore they are unbiased.

Given two parents A and B, the offspring are two new chains C and D such that a $A \cap B \subseteq C$ and $A \cap B \subset D$; a vertex $v \in A - B$ has a probability p^+ of being in C, and a vertex in $B - A$ has a probability p^- of being in C, where p^- is much lower that p^+. The set D is built the same way.

Mutation is defined as the random removing of a point of the chain, different from the first or last one.

3) Multiobjective fuzzy fitness: The set of objectives used in this approach are the number of uncovered fuzzy input data, and the total area of the polygonal chain. Both criteria are fuzzy numbers. This means that it is needed an operator *less than* and an operator *less or equal than*, both defined for fuzzy numbers, so dominance could be evaluated. In [23] the Pareto dominance concept is extended to fuzzy dominance, and different levels of α-cut are used for each decision making process, using the concept known as α-dominance. In [15] it is proposed a fuzzy rule to determine the *degree of dominance of*

[1] This chain might include some extra points not covered by the input data, but this always would benefits the taxi, thus it is legally correct.

x over y, and another fuzzy rule to determine the *degree of been dominated of x by y*. Then, aggregating those rules by means of the max t-conorm a crisp rank of dominance is obtain for each individual x. In [13] a totally different approach is used. It defines a comparison between fuzzy numbers, so Pareto dominance could be used as stated in its definition. In [11] a generalization of the Pareto dominance concept is proposed. In that work, instead of using especial operators *less than* and *less or equal than*, fuzzy Pareto dominance is defined so the result of such redefinition is that decision surface is obtained.

In this work it was decided to use the α-dominance approach. Each criteria is characterized as a fuzzy number, and for different levels of α-cuts values of such criteria are computed. The two criteria used, as stated before, were the total area of the polygonal chain and the percentage of uncovered data, both of them have to be minimized. Hence, the dominance is evaluated for each α-cut, obtaining a dominance result for each α-cut.

B. Deterministic longest path estimation

Once the data is preprocessed, and we have obtained from the genetic algorithm the smallest set of vertexes that contain the true path, we need to evaluate its LUB.

For each α-cut of the fuzzy b-splines that contain the taxi trajectory, we get a polygonal set constructed with trapezoids, as can be seen in Figure 4. The motion direction is indicated by a thin dashed arrow. Each trapezoid vertex is denoted with a pair of integers, the ones at the left of the arrow have zero at first, the ones at the right have one at first. The other number is the step in motion sequence. The longest path at each step i goes through $(0, i)$ vertex or $(1, i)$. The set of vertexes that define the longest path, can be computed by exhaustive exploration of all possible combinations, but this is very expensive in terms of computational cost and proved impracticable in a realistic trajectory with 700 points, for instance. This problem has been studied in the area of Computational Geometry and is related with Longest Path with Forbidden Pairs [2], that is NPO PB-complete.

Because of this and given that in a realistic trajectory the changes of direction and the changes in distance between left and right vertex are limited due to the dynamics of the taxi, the geometry of the road and GPS behavior, we use a heuristic that is lineal in time with the number of vertex. The heuristic is based in the selection of convex vertexes: when a vehicle turns, the longest path goes through the exterior of the trajectory curvature. The convexity of a vertex is analyzed using the straight lines that rely on previous and next vertexes, the possible relative positions of the central vertex can be seen in figure 3, where convex vertex are marked with a small circle and the lines that pass through vertex $(0, i-1)$, $(0, i+1)$ and $(1, i-1)$, $(1, i+1)$ are drawn. From left to right and up to bottom, if both vertex are between the lines, both are concave. If only one is out of the lines, it must be convex. If both are out of the lines, may be both are convex (left) or may be one is concave and the other convex. In both cases, if the farthest one from the nearest line is chosen , then it is convex.

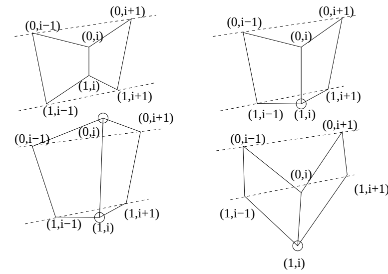

Fig. 3. Possible relative positions of vertex and lines between prior and next vertex.

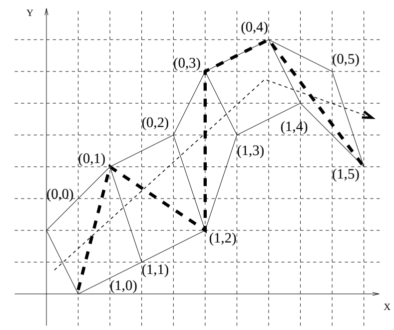

Fig. 4. Example of longest path estimation.

The heuristic is as follows: the first segment of the longest path goes from a convex vertex in step 1 to the vertex at step 0 that gives the maximum segment length. From step 1 to the one before the last, the path goes through:

- If there is only a convex vertex, trough this vertex.
- If there are two convex vertexes, through the farthest one.
- If there are no convex vertex, trough the farthest one.

The last segment ends in the farthest vertex from the previous one. In figure 4 the path computed with this heuristic is marked with a thick dashed line. The first segment goes from $(1, 0)$ to $(0, 1)$ because $(0, 1)$ is convex and the distance to $(0, 0)$ is shorter. Then the longest path continues to $(1, 2)$ because is the only convex. The same happens with $(0, 3)$ and $(0, 4)$. Finally, the path ends in $(1, 5)$ because is farther from $(0, 4)$ than $(0, 5)$).

IV. Experiments and Results

In the experiments presented here, the parameters of the NSGA-II algorithm are: 400 generations, 100 individuals in the population, 0.1 and 0.7 of mutation and crossover probabilities, $p^{+} = 0.7$ and $p^{-} = 0.01$. Each individual must cover a minimum of 85 percent of input data to be included in the Pareto front.

We have decided to evaluate our algorithm in a realistic path that covers the situations usually found when the MOT test of a taxi is done. This includes several turns, accelerations and decelerations, changes in number of available satellites and thus, changes in HDOP and CEP [22]. GPS longitude/latitude coordinates were translated to Universal Transverse Mercator northing/easting coordinates in order to easy distance calculations between GPS fixes [21]. With this system, points in earth

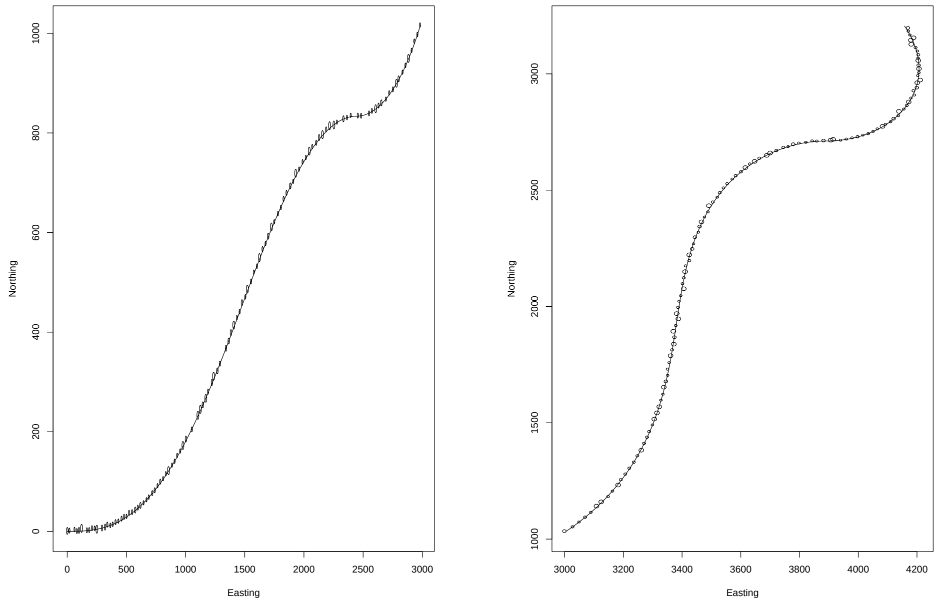

Fig. 6. Trajectories used in the experiments.

surface are projected on to a equal spaced planar metric grid, therefore the distance between fixes is the usual Euclidean one.

The trajectory is sampled each second, obtaining 1000 points, the total length of the trajectory is 21273.21 meters. At each location, we take a random number from 4 to 9 as the number of available satellites, that we found representative for real data. From this data, we build a dataset of GPS measures, sampled at each second. Each measurement is simulated using the following procedure, with a probability of 0.95, a point is selected that is closer in distance to the real one less than the CEP at that probability. With 0.05 probability the point is selected further than the corresponding CEP from the original data. This resembles the uncertainty that occurs using GPS, and the obtained data can be used to test how tight the bounds obtained with our algorithm are. The reader must remember that the goal is to obtain a multi polygonal chain that covers most of f the GPS fixes with minimum number of vertexes and with the minimum area. In figure 5 is shown part of the generated data. GPS measures are represented with circles (actually ellipsoids due to scaling issues) with radius equal to 95% CEP and the original trajectory with a continuous line. As can be seen, most of the circles intersect the trajectory, that is, most of the points of the real trajectory (in fact 95%) are inside the circles with CEP radius, centered in GPS fixes.

We perform two experiments with two subset of the complete dataset with 120 points each. They can be seen in the left (first) and right (second) sides of Figure 6.

The true length of the first trajectory is 3228.574 meters. The estimated length of the longest path compatible with the 85 % of the points of the first processed trajectory polygonal chain is 3418.81. If the taximeter reports a distance longer more than 10% than this upper bound, it should be rejected because even in the worst case the taximeter is out of tolerance. The distance through the GPS fixes is 3238.521, that is much closer to the real data, but the taxi owner can argue about the uncertainty of the procedure saying that it is inaccurate, if we compute an upper bound of the length compatible with GPS data there is no chance for this.

The length of the second trajectory is 2741.306 meters. The estimated length of the longest path compatible with the 85 % of the points of the corresponding processed trajectory polygonal chain is 3250.78. In this case the bound is less tight since the trajectory has stronger turns and this leads to longest path compatible with the data.

V. Conclusions and future work

During the development of this application we found that if we report directly the data obtained with GPS equipment, there were legality issues about the uncertainty of the measures. Taxi owners could easily gain in courts any reclamation where the uncertainty of the GPS measures where revealed. Nowadays the only system that we can use to certify a taximeter in a MOT (as stated in I) is to use a GPS device, so we have to modify the point of view of the MOT test procedure. Because of this we choose to give as result the upper bound of the trajectory length compatible with GPS data, in this way there is no doubt in that if the taximeter reported length is 10% above of this measure, then it should be rejected. Additionally, this alternative is less restrictive with the real data given the biased error detected in the taximeters. We have found that our algorithm performs worst when the trajectory includes more and stronger turns, this issue must be solved in future

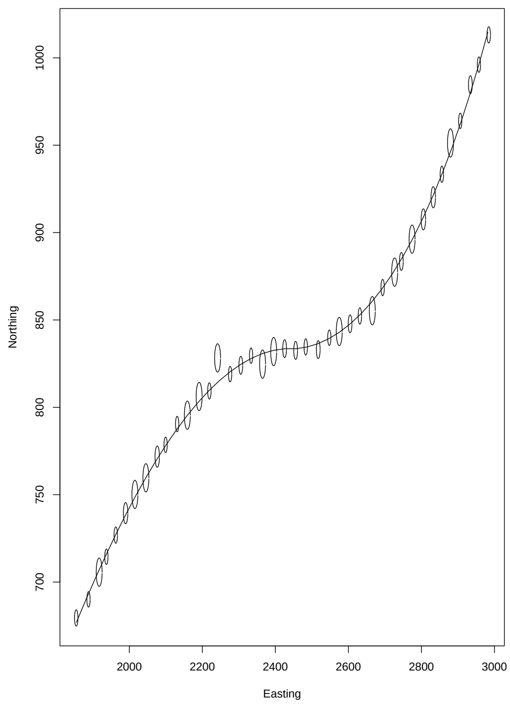

Fig. 5. Example of GPS generated data along with the real trajectory.

modifications with an additional heuristic that includes the dynamic behavior of a real driver.

Future work includes using different evolution algorithms as simulated annealing with genetic operators, which is faster and performs well in multi objective problems [19]. In the same way, different fuzzy dominance approaches should be tested to better fit the longest path better.

ACKNOWLEDGMENT

This work was funded by Spanish M. of Education, under the grant TIN2005-08386-C05.

REFERENCES

[1] A.M. Anile, B. Falcidieno, G. Gallo, M. Spagnuolo and S. Spinello, *Modeling uncertain data with fuzzy B-splines*, Fuzzy Sets and Systems 113, 397–410, 2000.

[2] Berman, P., and Schnitger, G. (1992), "On the complexity of approximating the independent set problem", Inform. and Comput. 96, 77-94.

[3] C. de Boor, *On Calculatin with B-splines*, Journal of Approximation Theory 6, 5O-62, 1972.

[4] C. A. Coello, *An Updated Survey of Evolutionary Multiobjective Optimization Techniques : State of the Art and Future Trends*, 1999 Congress on Evolutionary Computation, IEEE Service Center, 1999.

[5] I. Couso, S. Montes, P. Gil The necessity of the strong alpha-cuts of a fuzzy set International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 9-2, 249-262, 2001

[6] K. Deb, S. Agrawal, A. Pratab, and T. Meyarivan, *A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II*, In Marc Schoenauer, Kalyanmoy Deb, Günter Rudolph, Xin Yao, Evelyne Lutton, Juan Julian Merelo, and Hans-Paul Schwefel, editors, Proceedings of the Parallel Problem Solving from Nature VI Conference, 849-858, Springer. Lecture Notes in Computer Science, 2000.

[7] K. Deb and T. Goel, *Controlled Elitist Non-dominated Sorting Genetic Algorithms for Better Convergence*, In Eckart Zitzler, Kalyanmoy Deb, Lothar Thiele, Carlos A. Coello Coello, and David Corne, editors, First International Conference on Evolutionary Multi-Criterion Optimization, 67-81. Springer-Verlag. Lecture Notes in Computer Science No. 1993, 2001.

[8] D. Dubois and H. Prade, *Fuzzy elements in a fuzzy set*, Proceedings of 10th International Fuzzy Systems Association (IFSA) Congress, Beijing, Springer, 2005.

[9] Steven Dutch, "Converting UTM to Latitude and Longitude (Or Vice Versa)". http://www.uwgb.edu/dutchs/UsefulData/UTMFormulas.HTM. University of Wisconsin - Green Bay.

[10] R. Estkowski and J. S. B. Mitchell, *Simplifying a polygonal subdivision while keeping it simple*, SCG '01: Proceedings of the seventeenth annual symposium on Computational geometry, ISBN 1-58113-357-X, 40–49, ACM Press, New York, NY, USA, 2002.

[11] M. Farina and P. Amato, *Fuzzy Optimality and Evolutionary Multi-objective Optimization*, in Carlos M. Fonseca, Peter J. Fleming, Eckart Zitzler, Kalyanmoy Deb and Lothar Thiele (editors), Evolutionary Multi-Criterion Optimization. Second International Conference, EMO 2003, pp. 58–72, Springer. Lecture Notes in Computer Science. Volume 2632, Faro, Portugal, April 2003.

[12] Goodman, Nguyen. *Uncertainty Models for Knowledge-based Systems*. North-Holland. 1985

[13] M. Hapke, A. Jaszkiewicz and R. Slowinski, *Pareto Simulated Annealing for Fuzzy Multi-Objective Combinatorial Optimization*, Journal of Heuristics, 6(3), 329-345, August 2000.

[14] J. Hershberger and J. Snoeyink, *Speeding up the douglas-peucker line simplification algorithm*, in Proceedings of the 5thInternational Symposium on Spatial Data Handling, 134–143, Charleston, United States, 1992.

[15] M. Köppen, K. Franke and B. Nickolay, *Fuzzy-Pareto Dominance Driven Multiobjective Genetic Algorithm*, In Proceedings of the 10th IFSAWorld Congress (IFSA 2003), pages 450-453, Istanbul, Turkey, June, 2003.

[16] N. Meratnia and R. A. de By, *Trajectory representation in location-based services : problems and solutions*, in Proceedings of the 3rd IEEE Workshop on Web and Wireless Geographical Systems (W2GIS 2003) in conjunction with the Fourth International Conference on Web Information Systems Engineering (WISE), Rome, Italy, 2003.

[17] L. Ortega, *Apuntes de Geometría Computacional*, Universidad de Jaen, http://wwwdi.ujaen.es/̃lidia/topografia/apuntes.html

[18] L. Sánchez and I. Couso, *Advocating the use of Imprecisely Observed Data in Genetic Fuzzy Systems*, I International Workshop on Genetic Fuzzy Systems GFS-2005, 2005.

[19] L. Sánchez, J. Otero and J. R. Villar, *Boosting of fuzzy models for high-dimensional imprecise datasets*, in Proceedings of the Information Processing and Management of Uncertainty in Knowledge-Based Systems IPMU06, Paris, 2006.

[20] L. Sánchez and I. Couso, *Advocating the use of Imprecisely Observed Data in Genetic Fuzzy Systems*. IEEE Transactions on Fuzzy Systems (publication pending.)

[21] Snyder, J. P., "Map Projections Used by the U. S. Geological Survey", 2nd edition, Geol. Survey Bulletin 1532, 313 p., U. S. Government Printing Office, Washington, D. C., 1982.

[22] Wilson, D., "David L. Wilson's GPS Accuracy Web Page", http://users.erols.com/dlwilson/gps.html.

[23] J. Zhang, B. Pham, and P. Chen,*Fuzzy Genetic Algorithms Based on Level Interval Algorithm*, In Kazmierczak, E, Eds. Proceedings The 10th IEEE International Conference on Fuzzy Systems, pages pp. 1424-1427, Melbourne, Australia, 2001.