Research Article

Using Evolutionary Computation on GPS Position Correction

Jung Yi Lin

Department of Computer Science and Information Engineering, Chien Hsin University of Science and Technology, Jhongli, Taiwan

Correspondence should be addressed to Jung Yi Lin; jylin@ieee.org

Received 5 August 2013; Accepted 12 November 2013; Published 20 January 2014

Academic Editors: J. H. Sossa and H. Xu

Copyright © 2014 Jung Yi Lin. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

More and more devices are equipped with global positioning system (GPS). However, those handheld devices with consumer-grade GPS receivers usually have low accuracy in positioning. A position correction algorithm is therefore useful in this case. In this paper, we proposed an evolutionary computation based technique to generate a correction function by two GPS receivers and a known reference location. Locating one GPS receiver on the known location and combining its longitude and latitude information and exact poisoning information, the proposed technique is capable of evolving a correction function by such. The proposed technique can be implemented and executed on handheld devices without hardware reconfiguration. Experiments are conducted to demonstrate performance of the proposed technique. Positioning error could be significantly reduced from the order of 10 m to the order of 1 m.

1. Introduction

Global positioning System, GPS, has been successfully applied in various areas such as navigation, meteorology, military tasks, mapping, tour design, path tracking tools, and more [1]. Recently many mobile devices have been equipped with embedded GPS [2] such as tablet PCs and smart phones. They provide maps to help users not to lose their way or search the shortest route to their destination.

GPS receiver receives satellite signal from some of constellation 24 GPS satellites. Those satellites are controlled by the United States Department of Defense [2]. The position of a GPS receiver, u , can be derived by the pseudorange ρ of a satellite. Let s be a satellite located at (x_s, y_s, z_s) ; u is located at (x_u, y_u, z_u) ; ρ can be evaluated by

$$\begin{aligned}\rho &= \|s - u\| + ct_r + v_p \\ &= \sqrt{(x_s - x_u)^2 + (y_s - y_u)^2 + (z_s - z_u)^2} + ct_r + v_p,\end{aligned}\quad (1)$$

where c is the speed of light; t_r is the offset of the receiver clock; and v_p stands for a random noise that is expected to be zero.

Theoretically, the exact position of u (x_u, y_u, z_u) can be determined with given four error-free satellite coordinates.

Unfortunately, GPS positional accuracy is affected by many factors [3, 4] such as radio signal corruption, ephemeris error, satellite and receiver clock offset, multipath error, receiver measurement noise, satellite geometry measures, tropospheric delay, and ionospheric delay [5]. In general, due to those noises, GPS position accuracy is degraded to the order of 10 m [6].

Many techniques are proposed to improve GPS position accuracy. A commonly used technique is to use relative positioning [7]. Relative positioning methods, including static, rapid static, pseudokinematic, kinematic, and real-time kinematic [7–9], have proved their ability of improving GPS accuracy. In [7], Berber et al. claimed that pseudokinematic technique produces closest results, which could significantly reduce the error to 2 centimeters.

Differential correction is an effective method to improve GPS positional accuracy. A GPS receiver with such technique is called dGPS. A typical differential correction requires a reference stationary receiver at a known location [10, 11]. Figure 1 shows a typical scenario of the dGPS environment. The exact location information of reference stationary receiver is known. It receives GPS signals and calculates its position. Under the assumption that close GPS receivers suffer similar noises and after evaluating the difference between the exact known position information and the

FIGURE 1: An illustration of dGPS scenario. The precise location of the reference stationary receiver is known. A GPS receiver which requires correction accepts both signals from satellites and the stationary receiver.

FIGURE 2: A consumer-grade GPS receiver at a location which has exact known location.

calculated position information, the reference stationary receiver communicates with roving GPS receivers to correct their position information. dGPS can be used to eliminate affections of ionospheric and tropospheric delay, ephemeris error, and satellite clock error. However, when the error is due to multipath error, or poor satellite measurement geometry, the improvement effectiveness of dGPS technique is relatively low.

The main drawback of using dGPS technique is that reference stationary receivers are not common in many countries. Fortunately, many accessible places have been precisely measured for their geometry location. If a consumer-grade GPS receiver could be a reference stationary receiver, it is possible to simulate a dGPS environment. Given two GPS receivers, G_1 and G_2 , where G_1 is placed on a known location, L_1 , the location information obtained by G_1 could be used to correct G_2 . Such scenario is shown in Figure 2. In this paper, we will use two consumer-grade GPS receivers to construct the scenario. Instead of using survey-grade GPS receivers, which have high accuracy and have been applied correction techniques, consumer-grade GPS receivers could be more common for most of users.

A novel position correction technique is proposed in this paper. This technique is based on differential correction and genetic programming (GP) [12]. GP will be used to generate a correction function from NMEA information [13] derived from the GPS receiver at the known location and the GPS receiver which needs to be corrected. The receiver which

requires to be corrected will apply the function to obtain its corrected location information.

2. Layered Architecture Genetic Programming

Genetic programming [12, 14] is a research area of evolutionary computation. It has been proved that GP is capable of finding a solution efficiently. GP, like other techniques in evolutionary computation, generates possible solutions—in this case, correction functions—randomly for the given problem under given constraints. These solutions are called *individuals*. In this paper, individuals are represented as functional expressions. The fitness value which of an individual is used to measure the degree of the individual fitting with the given problem is determined by a predefined fitness function. The set with fixed size of individuals is named a *population*. In order to produce new solutions, genetic operators such as crossover and mutation are applied on selected individuals, called *parents*, to create offspring and mutant. Comparing the fitness degree of those offspring and mutant with parents, which have higher fitness value, will be kept as survived individuals. All survived individuals will replace the original population. A *generation* is finished once the original population is fully replaced. After a number of generations, evolutionary process completes and the individual with highest fitness is regarded as the result [14].

In this paper, we use the improved version of genetic programming called layered architecture genetic programming, LAGEP [14]. LAGEP is only usable with functional expression individuals. It utilizes the layer architecture to arrange populations. Populations in the same layer evolve independently. Once every population finishes evolutionary progress, the best individual of each population evaluates with its training instances, T , to generate a series of numerical results. The number of results is equal to $|T|$. Combining those values, a new training set T' having $|T|$ instances could be produced. Supporting that the number of populations in the layer is n , T' will be an n -dimensional training set. The final layer of LAGEP contains one population only. The individual produced by this population is the evolutionary result [14]. The flowchart of LAGEP is shown in Figure 3.

Training instances are constructed by raw information obtained from two GPS receivers and the known location. GPS receivers are capable of transferring different types of NMEA interpreted sentences [13]. In this work, we used GPGBA to represent position information, as shown in Table 1. The third, fifth, tenth, and twelfth field are symbols that can be harmlessly eliminated. The value of sixth field indicates GPS quality which is fixed. The thirteenth and fourteenth are usable when dGPS is available. The fifteenth is the checksum used to identify correctness of received data. In conclusion, 8 out of 15 fields can be removed. Two GPS receivers construct a 13-feature training instance after eliminating a redundant UTC time feature since those GPS receivers would have identical UTC time. Those features with longitude and latitude of the known location form a 15-feature

FIGURE 3: The flowchart of LAGEP.

TABLE 1: Fifteen fields of GPGGA sentence.

Number	Meaning
1	UTC of position
2	Latitude
3*	N or S
4	Longitude
5*	E or W
6*	GPS quality indicator 1: invalid 2: GPS fix 3: dGPS fix
7	Number of satellites in use
8	Horizontal dilution of position
9	Antenna altitude above/below mean sea level (geoid)
10*	Meters (antenna height unit)
11	Geoidal separation
12*	Meters (units of geoidal separation)
13*	Age in seconds since last update from differential reference station
14*	Differential reference station ID
15*	Checksum

* Removed features.

training instance, as shown in Table 2. The target value is either known latitude or known longitude to which we intent to correct GPS receiver as close as possible.

TABLE 2: Features of a training instance.

Number	Meaning
Target value	Known latitude/known longitude
1-7	\$GPGGA from GPS receiver 1 as shown in Table 1
8-13	\$GPGGA from GPS receiver 2 as shown in Table 1 without the UTC of position
14	Known latitude (where GPS receiver 1 is located)
15	Known longitude (where GPS receiver 1 is located)

An individual, idv , is defined as a functional expression composed of variables, operators, and constants:

$$\begin{aligned}
 idv &= (S_v, S_{op}, C), \\
 S_v &= \{X_i \mid i = 1, 2, \dots, 17\}, \\
 S_{op} &= \{+, -, \times, /, \sin, \cos, \log\},
 \end{aligned} \tag{2}$$

$$C = \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}.$$

An individual is a function mapping 17 real value features with constants into single real value, that is, $idv: (\mathbf{R}^{17} \cup C) \rightarrow \mathbf{R}$, which is supposed to be as close as the target value. The target value is the value what an individual is evolving for. When we attempt to acquire a correction function for latitude, the latitude information will be the target value during this run of the evolutionary process and is the only thing concerned by an individual.

A training instance, t , and the training set, T , are defined as follows:

$$\begin{aligned}
 t &= (\text{target value}, f_1, f_2, \dots, f_{15}), \\
 T &= \{t_i \mid i = 1, 2, \dots, |T|\};
 \end{aligned} \tag{3}$$

the target value is either known latitude or known longitude. The fitness of an individual is defined by

$$\text{fitness} = \sqrt{\sum_{i=1}^{|T|} (idv(t_i) - \text{target value})^2}, \tag{4}$$

where $idv(t_i)$ stands for the calculated valued of training instance t_i by the individual. Overfitting is a situation that a trained individual highly fits the training set but obtains relatively poor performance for the test set. To avoid the occurrence of such phenomenon, the validation process is applied. An individual having highest score is the output of the population:

$$\text{score} = \text{fitness} + \sqrt{\sum_{i=1}^{|V|} (idv(t_i) - \text{target value})^2}, \tag{5}$$

where $|V|$ is the number of instances in the validation set.

3. Experiments

Two public reference positions, UCH01 (24.94728, 121.22916) and UCH02 (24.94719, 121.22951), are provided by Chien Hsin

FIGURE 4: The image of UCH01 and UCH02 (image from Google with a proportional scale of 30 m).

University as shown in Figure 4. Satellite image extracted from Google Earth is shown in Figure 3. Two consumer-grade GPS, HOLUX GPSport 245 [15], are precisely placed on UCH01 and UCH02 for 24 hours to collect position information. After eliminating noisy data, the dataset contains 59,209 instances. We used 19737, 19736, and 19736 instances as training set, validation set, and test set, respectively. To reduce the conversion error in calculating longitude/latitude format, UCH01 and UCH02 are transformed into NMEA format (2456.8368, 12113.7496) and (2456.8314, 12113.7706), respectively:

$$\begin{aligned}
 24.94728 &:= 2400 + 0.94728 \times 60 = 2456.8368, \\
 121.22916 &:= 12100 + 0.22916 \times 60 = 12113.7496, \\
 24.94719 &:= 2400 + 0.94719 \times 60 = 2456.8314, \\
 121.22951 &:= 12100 + 0.22951 \times 60 = 12113.7706.
 \end{aligned} \tag{6}$$

In this paper, we conduct two experiments that use UCH01 and UCH02 to be the reference point and the target position in turns. We performed 5-fold cross-validation for 10 times to demonstrate the average performance. Settings used for GP are shown in Table 3.

Average distance errors between position information obtained by GPS receivers and the two fixed positions are considerable. Average error is in order of 10 meter, as summarized in Table 4. We also show the image of UCH01 and UCH02 and obtained position information by averaging 59209 instances in Figure 5. Obviously, position information obtained by GPS receivers is unstable and inaccurate.

The training phase of GP is time consuming. Training time records of experiments are summarized in Table 5. It requires about one hour completing one experiment. It seems that the training time is not acceptable in real scenario. However, it is difficult to have that much position information to be training instances in real scenario as well. The training time is affected by the number of training instances. Fewer training instances would greatly cost less training time.

FIGURE 5: The image of UCH01, UCH02 and the position obtained by GPS receivers (image from Google with a proportional scale of 20 m).

TABLE 3: Genetic programming parameters.

Parameter	Value
Number of populations	Layer 1: 4
	Layer 2: 2
	Layer 3: 1
Population size	Layer 1: 250 for each population
	Layer 2: 500 for each population
	Layer 3: 1000 total: 3000
Generations	200
Crossover rate	0.9
Mutation rate	0.1
Operators	+, −, ×, /, sin, cos, natural log
Constants	0.1, 0.2, 0.3, ..., 0.9, 1.0, π
Depth of an individual	8

TABLE 4: Average error between GPS receivers and reference positions.

	Average error in latitude	Average error in longitude
GPS receiver at UCH01 (2456.8368, 12113.7496)	0.0050057407	0.0208154402
GPS receiver at UCH02 (2456.8314, 12113.7706)	0.0077285210	0.0203315121

Before showing the experimental results of LAGEP, we demonstrate a simple correction method based on location information obtained by both GPS receivers. Since G_1 is placed right on UCH01, longitude and latitude of G_2 minus the difference between G_1 and UCH01 should be close to UCH02. Denote longitude and latitude information reported by GPS G_1 and G_2 as $G_{1\text{long}}$, $G_{1\text{la}}$, $G_{2\text{long}}$, and $G_{2\text{la}}$. The correct longitude and latitude of UCH01 and UCH02 are denoted as $\text{UCH01}_{\text{long}}$ and UCH01_{la} and $\text{UCH02}_{\text{long}}$ and UCH02_{la} .

TABLE 5: Training time (in second).

Experiments	Target value: 2456.8368	Target value: 12113.7496
Experiment 1	3300.036	3878.202
Experiment 2	2588.838	4169.452
Experiment 3	2965.790	3083.224
Experiment 4	2676.224	3596.150
Experiment 5	2747.012	3520.266
Experiment 6	2364.806	4144.188
Experiment 7	2259.692	3614.860
Experiment 8	2433.826	3692.286
Experiment 9	2425.836	3876.082
Experiment 10	3137.874	3056.296
Average	2689.993	3663.101
Experiments	Target value: 2456.8314	Target value: 12113.7706
Experiment 1	2310.404	4091.588
Experiment 2	2323.656	3270.206
Experiment 3	2271.540	2818.054
Experiment 4	1891.328	4092.478
Experiment 5	1810.772	3855.086
Experiment 6	1856.274	3992.694
Experiment 7	2531.930	3416.518
Experiment 8	2775.832	4032.888
Experiment 9	2767.734	3770.812
Experiment 10	2581.238	3746.426
Average	2312.071	3708.675

TABLE 6: Average error on correct 1.

	Average error in latitude	Average error in longitude
GPS receiver at UCH01	0.0101397676	0.0411469523
GPS receiver at UCH02	0.0101397676	0.0411469523

Location information of G_2 , denoted as $G_{2\text{long}}'$ and $G_{2\text{la}}'$, is corrected by G_1 using

$$\begin{aligned} G_{2\text{long}}' &= G_{2\text{long}} - (G_{1\text{long}} - \text{UCH01}_{\text{long}}), \\ G_{2\text{la}}' &= G_{2\text{la}} - (G_{1\text{la}} - \text{UCH01}_{\text{la}}). \end{aligned} \quad (7)$$

G_1 is corrected by G_2 using

$$\begin{aligned} G_{1\text{long}}' &= G_{1\text{long}} - (G_{2\text{long}} - \text{UCH02}_{\text{long}}), \\ G_{1\text{la}}' &= G_{1\text{la}} - (G_{2\text{la}} - \text{UCH02}_{\text{la}}). \end{aligned} \quad (8)$$

The average error is shown in Tables 6 and 7. The corrected positions are shown in Figure 6. The corrected method seems reasonable but is inaccurate.

Experiment results of LAGEP on test sets are shown in Table 6. The average corrected position is close to target position with less than 1 meter. It demonstrated that the proposed method achieved significant result in both latitude and longitude. The degree of correction is significant. Standard deviation of those experiments shows that the experimental

TABLE 7: Average error of test set.

Experiments	Target value 2456.8368	Target value 12113.7496
Experiment 1	$3.60E - 11$	$2.03E - 06$
Experiment 2	$7.98E - 08$	$8.69E - 07$
Experiment 3	$2.74E - 08$	$2.49E - 06$
Experiment 4	$1.27E - 06$	$2.33E - 07$
Experiment 5	$4.61E - 06$	$2.73E - 06$
Experiment 6	$1.09E - 09$	$5.02E - 07$
Experiment 7	$1.90E - 07$	$2.50E - 06$
Experiment 8	$1.71E - 06$	$1.01E - 08$
Experiment 9	$1.46E - 08$	$4.41E - 06$
Experiment 10	$1.84E - 07$	$5.20E - 06$
Average	$8.09E - 07$	$2.10E - 06$
Standard deviation	$1.39E - 06$	$1.66E - 06$
Experiments	Target value 2456.8314	Target value 12113.7706
Experiment 1	$2.50E - 11$	$5.12E - 07$
Experiment 2	$0.00E + 00$	$1.48E - 06$
Experiment 3	$2.00E - 12$	$1.16E - 07$
Experiment 4	$0.00E + 00$	$3.21E - 06$
Experiment 5	$0.00E + 00$	$3.11E - 06$
Experiment 6	$4.49E - 09$	$6.13E - 07$
Experiment 7	$4.00E - 12$	$1.10E - 07$
Experiment 8	$4.00E - 12$	$1.64E - 06$
Experiment 9	$1.00E - 12$	$5.02E - 07$
Experiment 10	$1.00E - 12$	$2.89E - 08$
Average	$4.53E - 10$	$1.13E - 06$
Standard deviation	$1.35E - 09$	$1.14E - 06$

FIGURE 6: The image of UCH01, UCH02 and the corrected positions using simple correction method (image from Google with a proportional scale of 30 m).

results are stable. We illustrate the corrected positions in Figure 7. The corrected positions are almost overlapping with UCH01 and UCH02.

4. Conclusion

In this paper, we proposed a new GPS position correction technique based on layered genetic programming and the concept of dGPS. Experiments have shown that even when two GPS receivers have high error and noise, the proposed technique is capable of finding correction function to help find accurate position information. The proposed technique

FIGURE 7: The image of UCH01, UCH02 and the corrected positions using proposed method in black squares (image from Google with a proportional scale of 20 m).

could be easily implemented on mobile devices because it does not need to modify or install any hardware component. Our future work will focus on training correction function with time limitation constraints. The training phase stops when given time limitation is reached. Such would be closer to real world.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The author would like to acknowledge the assistance of Dr. Ming Chih Tung, Chia Hui Chang, Chao Chung Liu, and Ju Fu Peng for providing experiment equipments and collecting GPS information. The author also would like to thank anonymous reviewers who gave valuable suggestion that has helped to improve the quality of this paper.

References

- [1] L. E. Frenzel, "GPS takes a global position in the portable market," *Electronic Design*, vol. 55, no. 10, pp. 47–54, 2007.
- [2] A. El-Rabbany, *Introduction to GPS: the Global Positioning System*, Artech House, Boston, Mass, USA, 2002.
- [3] T. K. Yeh, C. S. Wang, C. W. Lee, and Y. A. Liou, "Construction and uncertainty evaluation of a calibration system for GPS receivers," *Metrologia*, vol. 43, no. 5, pp. 451–460, 2006.
- [4] J. Zhang, K. Zhang, R. Grenfell, and R. Deakin, "GPS satellite velocity and acceleration determination using the broadcast ephemeris," *Journal of Navigation*, vol. 59, no. 2, pp. 293–305, 2006.
- [5] J. A. Klobuchar, "Ionospheric effects on GPS," in *Global Positioning System: Theory and Applications*, vol. 2, pp. 485–516,

American Institute of Aeronautics and Astronautics, Washington, DC, USA, 1996.

- [6] L. L. Arnold and P. A. Zandbergen, "Positional accuracy of the Wide Area Augmentation System in consumer-grade GPS units," *Computers and Geosciences*, vol. 37, no. 7, pp. 883–892, 2011.
- [7] M. Berber, A. Ustun, and M. Yetkin, "Comparison of accuracy of GPS techniques," *Measurement*, vol. 45, no. 7, pp. 1742–1746, 2012.
- [8] C. D. Ghilani and P. R. Wolf, *Elementary Surveying—An Introduction to Geomatics*, Prentice Hall, Upper Saddle River, NJ, USA, 12th edition, 2007.
- [9] J. Van Sickle, *GPS For Land Surveyors*, CRC Press, New York, NY, USA, 3rd edition, 2008.
- [10] M. Chivers, *Differential GPS Explained*. ArcUser, 2003.
- [11] P. Bolstad, A. Jenks, J. Berkin, K. Horne, and W. H. Reading, "A comparison of autonomous, WAAS, real-time, and post-processed global positioning systems (GPS) accuracies in northern forests," *Northern Journal of Applied Forestry*, vol. 22, no. 1, pp. 5–11, 2005.
- [12] J. R. Koza, *Genetic Programming: on the Programming of Computers by Means of Natural Selection*, MIT Press, Cambridge, Mass, USA, 1992.
- [13] National Marine Electronics Association (NMEA), <http://www.nmea.org/>.
- [14] J.-Y. Lin, H.-R. Ke, B.-C. Chien, and W.-P. Yang, "Designing a classifier by a layered multi-population genetic programming approach," *Pattern Recognition*, vol. 40, no. 8, pp. 2211–2225, 2007.
- [15] HOLUX Technology Inc, GPSport 245, http://www.holux.com/JCore/en/products/products_content.jsp?pno=349.