Improvements In Dynamic GPS Positions Using Track Averaging

James R. Clynch
Naval Postgraduate School
Department of Oceanography
Code OC/Cl
Monterey, CA 93943

Richard Franke
Naval Postgraduate School
Department of Mathematics
Code MA/Fe
Monterey, CA 93943

Beny Neta
Naval Postgraduate School
Department of Mathematics
Code MA/Nd
Monterey, CA 93943

August 30, 1999

Contents

List of Figures

List of Tables

Abstract

The issue of improving a Global Positioning System (GPS), Precise Positioning System (PPS) solution under dynamic conditions through averaging is investigated. Static and Dynamic data from the Precision Lightweight GPS Receiver (PLGR) were used to analyze the error characteristics and deign an averaging technique for dynamic conditions.

It was found that the errors in PPS solutions are dominated by the satellite broadcast ephemeris parameters. The solution errors are highly correlated for a given set of satellites/ephemeris. The variation can be as low as 0.4 m in dynamic conditions, but a slowly changing "bias" of several meters is also present.

For fitting the location of a road observed repeatedly with a PPS receiver a technique based on "space curves" was developed. Here the solutions are transformed from functions of time to functions of space (location). These then are used. Curves could be fit with a Bezier polynomial easily to the 0.4 m level. These analytic curves were then used to form an ensemble average. The bias vectors between the solutions were found with least squares estimation. These vectors were averaged using several techniques. This idea was applied to a short rad segment. Using 9 independent measurements taken over 6 months, the road was surveyed at the submeter level.

DoD Key Technology Areas: Ground Vehicles, Computing and Software, Sensors.

Keywords: GPS, Global Positioning System, Dynamic Positioning.

1 Introduction

The accuracy of a GPS receiver in the Precise Positioning Spectrum (PPS) is on the order of 5 m horizontal and 7 m spherical today (1999)[1]. While this may be adequate for some applications, there are others that need somewhat better positions, but not as good as a survey position. In theory averaging independent PPS position estimates an do this. For static positions this seems simple, but there are some complications hidden in the independence of position estimates made with GPS. In addition if the needed information is the track of a road, things are much more complex. This study has attempted to address the issue of how to effectively average GPS PPS positions to achieve better location estimates in both the static and dynamic conditions. The emphasis will be on the dynamic case as it is the more difficult.

Here absolute, standalone, positions are considered as the raw input data for further processing. Clearly higher accuracy can be obtained through the use of differential GPS, but the focus here is what can be done with the absolute positions that come from PPS receivers. In particular the work will focus on the Precision Lightweight GPS Receiver (PLGR) which is very common (over 100,000 delivered) in the US military. This receiver uses 4 GPS range measurements to compute a position. It is a single frequency receiver, which limits its height accuracy somewhat. These results will be a floor on what could be achieved with better PPS receivers with more channels and/or dual frequency tracking.

In the case of the static receiver, the position solution can be significantly improved only by averaging very long periods, on the order of a day. The results of both a long period static result and a stop and go experiment will be presented. Repeated revisits to a site within an hour did not significantly add information unless the satellite set being tracked had changed.

For dynamic cases the route must be repeatable, at least at the 1 to 2 meter level in order to successfully combine solutions. The averaging of dynamic solutions is achieved by converting the tracks from time histories to tracks in space. In this study the tracks are computed in the two horizontal dimensions. The third dimension can be added later through various methods. The procedure for generating the space tracks involves selecting fairly short tracks and finding the corresponding data in multiple data sets. Each is converted to a parametric polynomial in space. A Bezier representation is used. This is essentially a piecewise cubic fit with continuous values and continuous first derivative. The latter is important because the normal to the curve is used in the process of combining curves to find an average track.

A system to locate a road using a database of PPS positions is diagramed in Figue 1. Here an operator identifies the road or feature to be geolocated. This could be a graphical interface or an area defined by geographic coordinates. The program would select the tracks of data that fit the operator's criteria. These tracks are the input data to the techniques described here. In the current study, the selection phase will not be addressed.

The first step in the process is the conversion of the tracks from functions of time to a function of spatial coordinates. These are the "space curves" that are analyzed further. The individual instances will be called track segments. The space curves chosen here are the Bezier representation.

It is assumed that the track segments differ from each other by a constant bias vector.

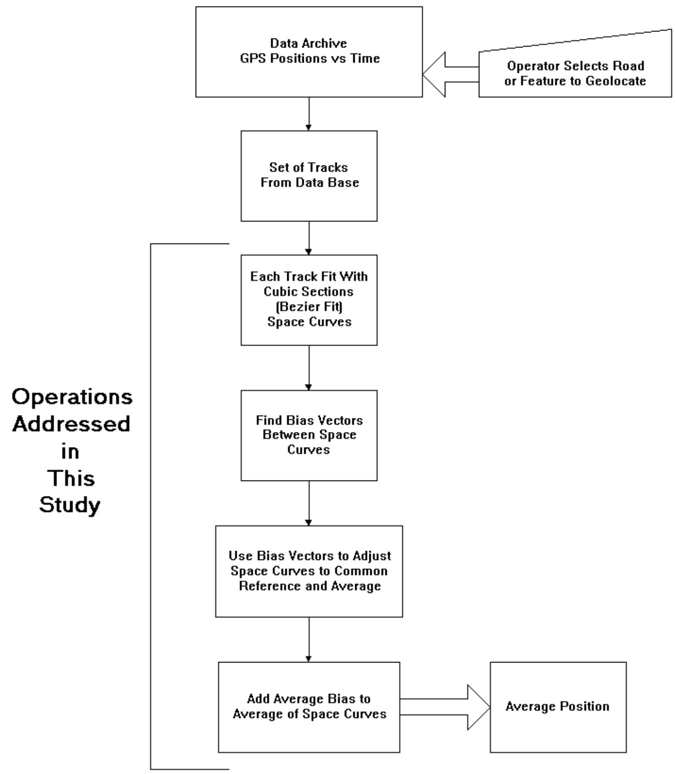

Figure 1: Diagram of Track Averaging

This is an assumption that is validated with experimental data in the study. The assumption depends on the same set of satellites being tracked during the time that the track segment is measured and that the time interval of he measurement is short (a few minutes or less).

The biases between all track segments can be computed in a least squares process. These biases can then be averaged directly or in a weighted manner. A method used in the analysis of atomic clocks (N-Cornered-Hat) is used to find the effective noise in each track compared to the ensemble, before the ensemble is formed. This allows not only correct weighting, but the editing of outliers due to satellite changes or many other factors. The tracks can be moved together using the bias vectors between one track and the others. This can be averaged. The net bias of this ensemble is the negative of the average of the biases between tracks.

The following sections will describe in detail the underlying assumptions made in this technique. These were illustrated by previous data taken on a ship. Here new data is taken with PLGR's under both static and dynamic conditions. Dynamic data was taken repeatedly over three of these areas. The data from one was used to illustrate the process of dynamic track averaging.

After a general background laying out the assumptions in chapter 2, the mathematical approach to the problem is developed in chapter 3. The test data is described in chapter 4. A detailed mathematical description of the analysis is presented in chapter 5. The data is applied to space tracks in chapter 6, which is the heart of the analysis technique. Finally, a specific dynamic example is analyzed with this technique in chapter 7. Submeter positioning of short road is demonstrated.

2 Background

2.1 Errors in PPS Range Measurements

The error in a GPS absolute position is roughly the Dilution of Precession (DOP) times the range error standard deviation. Therefore an understanding of the errors in a range measurement is needed. A diagram of the components of a range error is shown in Figure 2. Here the range to the satellite will be on the order of 20,000 km. The receiver clock error, while large, is estimated with each position and does not have a dominant effect on the solution error. The errors that are important, included in the "other" category on the top line, are expanded on the second line.

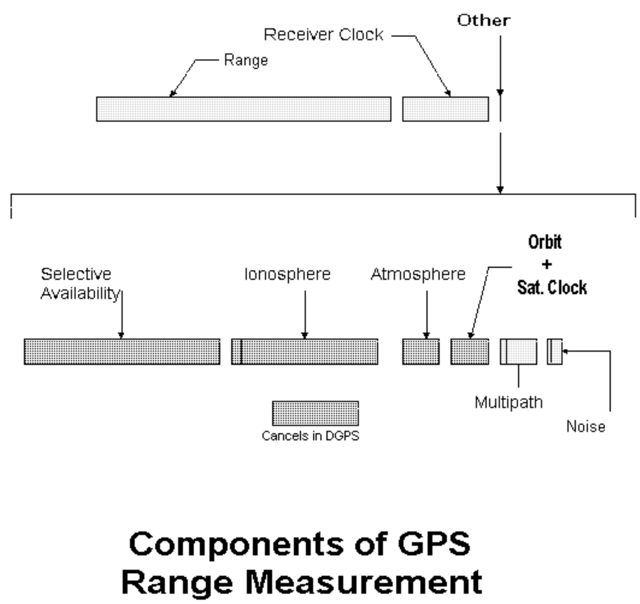

Figure 2: Components of GPS Range Measurements

For the military user in PPS mode, the Selective Availability (SA) error is removed in the receiver. For dual frequency receivers the same is true for the ionospheric error. While the PLGR's used here are single frequency and suffer from this error, its effects are mainly in the vertical component. The small vertical bar indicates the minimum ionospheric error. For reference the largest ionospheric error shown here is about 30 m. The sizes in this diagram are only approximately to scale.

The atmospheric error also affects mainly the vertical component. It can also be modeled quite accurately with just knowledge of altitude, at least at the 25 cm level or better. The last two components are dependent on the receiver and its environment. They usually vary rapidly, especially in a moving receiver, and can be easily averaged down. They will not be considered further here.

The other component, Orbit and Satellite Clock, is the most important for the PPS user. In order to find a position from GPS ranges, the receiver must know the location of the satellites at the time the signal was sent. This is done through a model of the satellite position. The parameters for this model are broadcast along with the ranging information by each satellite. In addition, the offset of the spacecraft clock from an absolute time system is included in the parameters broadcast. This is necessary because the GPS ranges are found by subtracting the transmit time from the received time and multiplying by the speed of light. This is about 30 cm (or a foot) per nanosecond (1/1000 microsecond or one billionth of a second.) Clearly timing errors are important. This is why the receiver clock offset is computed as part of each and every solution. The satellites have atomic oscillators, but even these wander over the course of a day by a few nanoseconds.

It is the inaccuracy in these parameters that the satellites broadcast to the user (commonly called the broadcast ephemeris or broadcast message) [1] that dominates the military users' PPS solution error. It is felt that the satellite clock parameters are dominant in this parameter set. These errors occur because the broadcast message numbers are projections of what will be, not measurements of what has been.

The GPS Operational Control Segment (OCS) measures the satellites' positions and clock state every 15 minutes from 5 ground monitor stations scattered throughout the world. (It is planned to add the National Imagery and Mapping Agency (NIMA) 5 ground stations to this network in the near future bringing the number of ground stations available to the OCS to 10 or more [1].) While the OCS computation center may have a good idea of the satellite parameters, this estimate is not what the user sees. Once or twice a day a set of model parameters for the future few days is prepared and sent up to each satellite. These are stored in an onboard memory and are broadcast to the user. Normally these projections never get more than 24 hours old. But that means that the information used in position computation is based on measurements made an average of 12 hours ago.

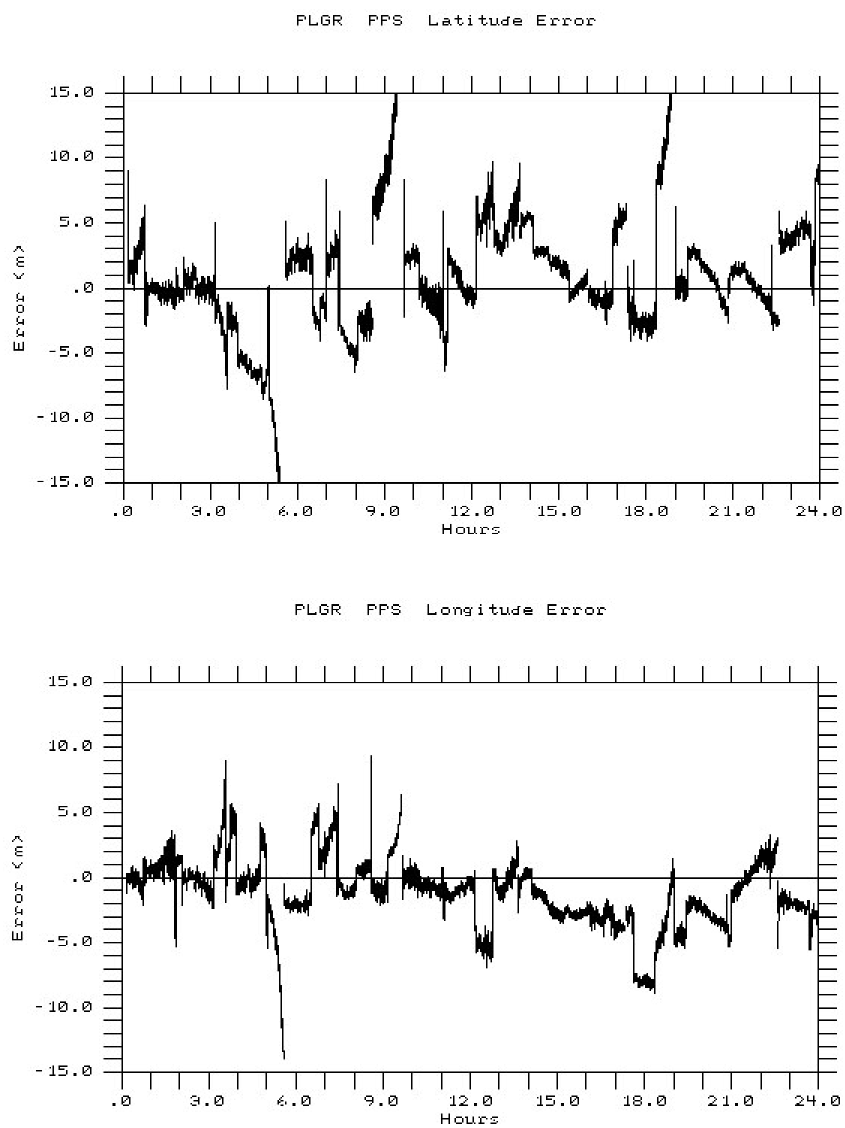

Figure 3: PLGR PPS Error While Static

2.2 Errors in PPS Real Time Positions

The difficulty in projecting the satellite states, particularly the onboard atomic clock error, is the principal cause of the orbit and satellite clock error. (This is really the combined radial and clock error, but will be called the clock error here.) This error will be effectively uncorrelated between satellites. It will also approximately be random between upload parameter sets. However it will be a slowly varying function of time for each satellite within a given upload.

If a receiver tracks the same set of satellites for several minutes, the error in position will be approximately constant. This is because the orbit and clock error from each of the satellites tracked will be almost constant over that time frame. However if the receiver changes the satellites it is using in its position computation it will be changing one of these errors for another. Even for the substitution of one satellite this can cause the position to jump by several meters. It will remain at that new level until another satellite change occurs.

An example of this behavior can be seen in Figure 3. Here the latitude and longitude errors are plotted from PLGR solutions on a fixed site over one day in mid 1997. The data was taken every second. Clearly these errors are not independent random variables on the time scale of 1 second. The errors look like constants over time intervals of a few minutes and a straight line over some periods of an hour. On top of this behavior is some noise, but more significantly jumps. The linear segments occur during the tracking of a fixed set of satellites. The errors are not constant because the contribution of each satellite error to the position errors changes as satellite geometry changes. The jumps occur when satellite sets change.

Clearly some changes of satellites have larger effects than others. While the DOP is always improved when these receivers chose to change satellites, sometimes the error increases. Examples of this in Figure 3 occur at about 9 hours and 18 hours. The difficulty is that the receiver has no knowledge of the error on any particular satellite. The size of individual errors is believed to arise mainly from the age of the data used in the broadcast ephemeris. This is essentially the time since last upload.

3 Approach

For this study new data were collected on several roads near the Naval Postgraduate School. These data were converted to a local cartesian coordinate system with the x-axis east west and the y-axis north south. The height was carried along as is. A kinematic reference trajectory was generated in each case. The cartesian data were then analyzed to generate a curve in space, thus removing the dependence on the time the data was collected. These space curves were then combined to generate average location for the roads.

The next subsection will outline the processing techniques. Addressing data acquisition in general will follow this. Detailed analyses follow.

3.1 Mathematical Approach

3.1.1 Overview

In order to average approximate paths, one has to first identify data from track segments of interest. At this time, the identification process, including a check for independence, is done by hand, with some automation. We will discuss this in Section 6. Once independent track segment data sets are found, an analytic representation for each track is obtained using some form of approximation. This step is discussed in the next subsection. This step will create for each track segment an analytic representation of the track segment for each data set. The averaging process for these approximations will be discussed in section 7.

3.1.2 Parameterization

In many computer-aided geometric design problems, one wishes to produce a smooth curve from a given ordered set of data points. Here we are given a set of points describing a curve in space in parametric form. The natural parameter in this case is time. With a parametric fit, each of the coordinates is fit as a function of the parameter, with the path then being traced out as the parameter varies.

While the natural parameter in this case is time, with such a parameterization it is difficult to combine data from multiple trips along the same path. Some authors have suggested the use of chord length spacing (Euclidean distance between points) because it approximates the arc length of (or distance along) the curve [2]. A number of other possible parameterizations could be used [3]. There is no "best" parameterization since most known methods can be defeated by a suitably chosen data set.

The methods employed by the two referenced papers and most other authors involve fitting cubic splines to the data. This can be done in at least two ways: attempting to minimize the distances from the data to the curve at fixed parameter values (a linear problem once the parameterization has been fixed), and attempting to minimize the distances from the data points to the curve. In the latter case, the actual parameter values of the nearest points on the curve must be discovered as part of the fitting process, and thus this is a nonlinear problem. While the linear problem is far easier to solve, the results cannot be as good because of the necessity to assume the parameterization a priori. Therefore we have chosen to fit curves to the data by minimizing the sum of the distances from the data points to the curve. This is called "Orthogonal Distance Regression", or ODR [4].

There are many possible forms that can be assumed for the fitting function. While polynomials naturally come to mind, they often exhibit poor fitting properties and might require excessively high degrees. Piecewise polynomials are usually a better choice, and there is a considerable literature on the topic. Cubic splines are the choice of most authors.

The use of cubic splines is desirable because splines are well known for their superior fitting properties. The parameters that define the spline, however, must satisfy a number of constraints (the continuity of value, slope, and curvature) making it difficult to specify the problem in such a way that the defining parameters are independent, a desirable trait for optimization. In addition, because we are modeling roadways, the large values of curvature at corners will pose a problem for curves with continuous curvature. Therefore, in our approach we have relaxed the smoothness conditions to require only continuity of the slope between cubic pieces (usually; in fact the form adopted may automatically incorporate corners if the data warrants it). A set of Bezier curves fitting a data set generate a curve that is continuous and has continuous first derivatives even at the connecting points (called knots). The description of Bezier curves typically takes a geometric flavor. Four control points define a single Bezier cubic curve (in two dimensions) $p_i = (x_i, y_i), i = 0, 1, 2, 3$, and the curve is given by

$$x(t) = (1-t)^3 x_0 + 3(1-t)^2 t x_1 + 3(1-t)t^2 x_2 + t^3 x_3 \quad 0 \leq t \leq 1$$

$$y(t) = (1-t)^3 y_0 + 3(1-t)^2 t y_1 + 3(1-t)t^2 y_2 + t^3 y_3 \quad 0 \leq t \leq 1.$$

The three line segments connecting the control points, form an open polygon called the control polygon. An example of a single Bezier curve is shown in Figure 4, and the parameters are described in the sidebar. More information can be found concerning Bezier curves in Gerald Farin [5]. Note that the curve starts and ends at the point p_0 tangent to the first polygon side and ends at p_3 tangent to the last polygon side. The curves will not ordinarily pass through the other two control points. The example demonstrates the relationship between the control polygon and the curve, illustrates the tangency properties, and the basic propensity of the curve to follow the control polygon.

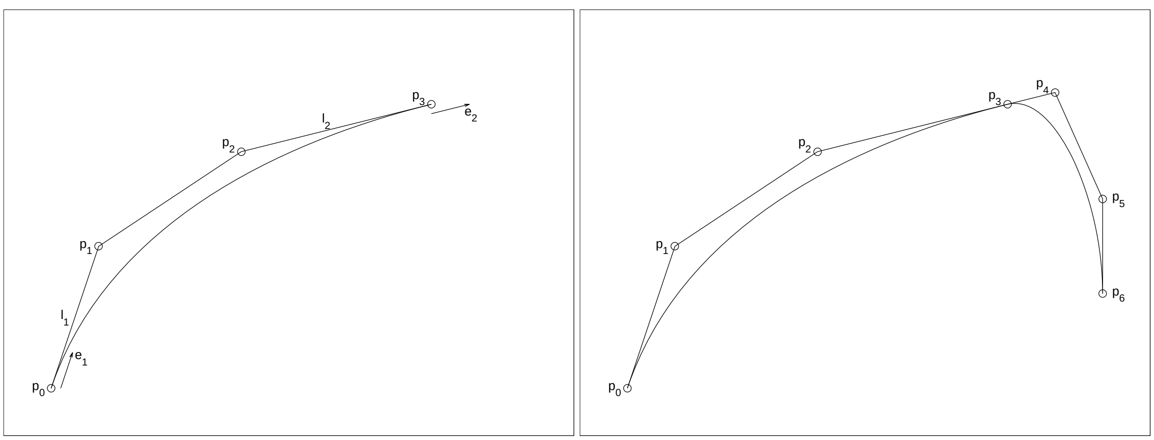

Figure 4: Bezier Segments Showing Notation. (a) One Segment Bezier, (b) Two Segment Bezier

The parameters are shown for one cubic segment in Figure 4 (left). The eight parameters for this segment are

end points	$p_0 = (x_0, y_0)$	2 Parameters
	$p_3 = (x_3, y_3)$	2 Parameters
end directions	$e_0 = (\cos(\theta_0), \sin(\theta_0))$	1 Parameter
	$e_3 = (\cos(\theta_3), \sin(\theta_3)$	1 Parameter
distances to interior	ℓ_1	1 Parameter
control points	ℓ_2	1 Parameter

Thus, $p_2 = p_1 + \ell_1 e_1$ and $p_2 = p_3 - \ell_2 e_2$.

In joining following segments, p_3 and e_3 become p_0 and e_0, respectively, of the following segment. Thus there are 5 parameters for each continuing segment.

With this control structure it is easy to concatenate two or more cubic segments joining with continuous slope. Because of the tangency condition that is satisfied, the curve may be extended. The continuous slope provided the first control point of the next segment coincides with the last control point of the current segment. The second control point of the second segment is on the line joining the last two control points of the current segment. The right part of figure 4 shows how a second cubic segment joins with continuous slope at the point p_3. The curve is easily extended to any number of segments.

The initial work in implementing these ideas was by M. R. Holmes in his M.S. thesis [6]. He developed Matlab software to solve the problem in two dimensions. The algorithm was further developed by E. Lane [7]. The independent parameters that determine the Bezier curve are the locations of the knot points, the directions of the unit tangent vectors at the knot points, and the location of the inner control points. These inner control points, p_1 and p_2, are constrained to lie on the line containing the unit tangent vector at the adjacent knot and at specified distances from the knot points, (see Figure 4). This ensures a curve with continuous slope between adjoining cubic segments, called G^1 continuity.

The problem of finding an optimal set of parameters is nonlinear, hence it is difficult to find the actual global minimum. On the other hand, with good initial estimates of the solution, good approximations can be found with a reasonable amount of computation. The current version uses a fixed number of knot points, decided a priori, although software is available that allows the insertion of additional knots (exactly duplicating the existing curve) and the deletion of knots (giving a new approximate curve). The final positions of the control points are found in an optimization process using these initial values.

In the previously mentioned theses [6, 7], it was assumed the data was given as ordered. This was important in that no assumption was made regarding whether a curve could cross itself (and in fact, this happened in the examples given). Since the ordering was given, it was then possible to determine which of two crossing segments of the curve a nearby data point was close to in the parametric sense, not just the geometric sense. While it may not be possible to easily order the data a priori in this application, knowing that the curve does not cross itself will enable us to determine the ordering of the points from multiple passes during the fitting process.

The process of fitting the track segments with a Bezier curve takes place in three steps. First an initial guess for the control points is made. This currently is done in a semi-automated fashion. The optimization is carried out in two phases. The first is a local

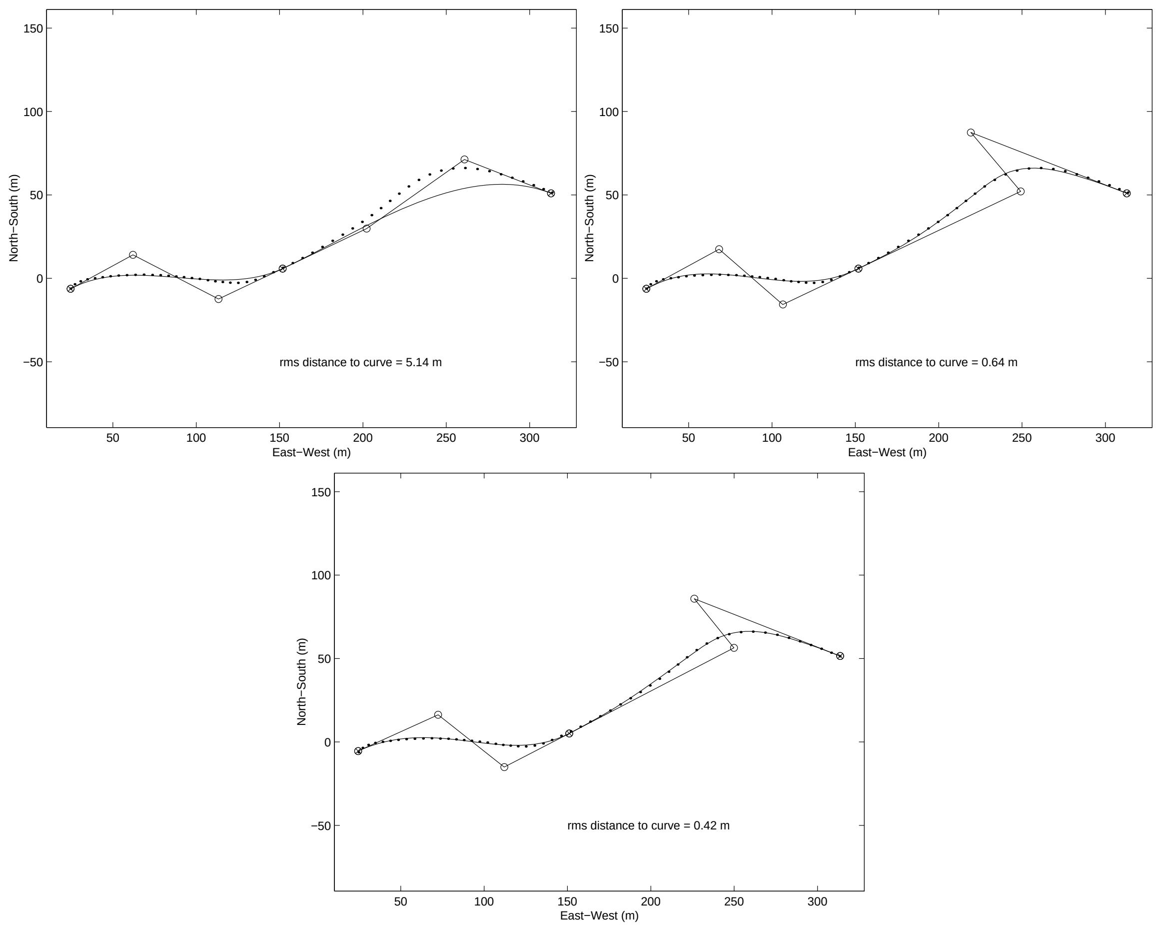

Figure 5: Control Points of Bezier Curve at 3 Stages of Optimization. (a) Initial Guess, (b) Local Optimization, (c) Global Optimization

optimization for the location of the interior control points located on the lines tangent at the knots. This is followed by a global optimization for all the parameters of the Bezier segments. For the purpose of this study, the optimizer built into Matlab (version 5.2) via its FMINS function. This uses a Nedler-Mead simplex (direct search) method. As an alternate the Matlab optimization toolbox function FMINU was also investigated. This uses the BFGS Quasi-Newton method. While the solutions were not identical, the produced essentially the same space curve.

Figure 5 shows an application with two cubic segments. The data on which this example is based was taken at the "beach lab", and consists of 54 points. The left figure represents the control polygon and the approximating curve after the user has input the initial guess knot points. The program then determines tangent vectors at the knots and distances to the interior knot points. The rms distance of the data points from the curve in Figure 5 is 5.14 m. The center figure shows the approximating curve and knot points after local optimization for placement of the interior knot points, with no changes to the location of the knot points, or the slopes at the knot points. The configuration of the right control polygon shows the flexibility of the method to adapt to move complicated shapes. The rms distance to the

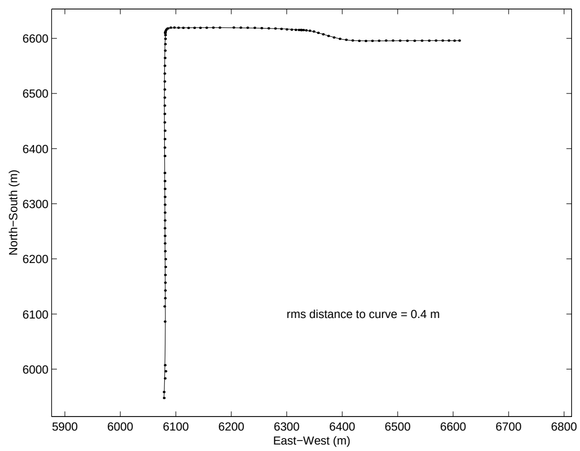

Figure 6: FT Ord Square Area Fit with Three Segment Bezier

curve for this case is about 0.64 m. The right figure shows the control polygon and the fit after all parameters have been optimized. The rms distance to the resulting curve is about 0.42 m. The large reduction in error after the local optimization and the relatively small error reduction after the global optimization reflects some skill by the user in proper initial placement of the knot points.

Another example is shown in Figure 6. Here the data consists of 99 points that were fit using a three segment curve. Recall that such a curve embodies a total of 13 parameters. The rms of the distances from the data points to the curve in this approximation is 0.40 m. The data was taken on a trip along the west and northern sides of the Ft. Ord square, which includes the kink previously noted. The path essentially consists of 3 nearly straight-line segments, joined by a sharp corner and by a transition (kink) from one line to another. While the control polygons and knots are not shown, the interior knots are near the corner and the midpoint of the kink. This example illustrates the capability of the fitting procedure to model very different kinds of behavior, from small radius corners to smooth transitions between essentially straight lines. To achieve the small radius corner the algorithm places the adjacent interior control points close to the knot at the corner.

Thus in two cases, one fairly extreme, this approach fit the data at the 0.40 m level. This is consistent with the differences in the zero baseline experiment on shipboard given in Table 1. It would probably not be useful to try and fit the raw data more accurately.

The Bezier curve fits discussed here assume random noise with zero mean. However the true non-random nature of the noise will then be folded into the process. As we discuss later, it is useful to separate segments with fixed satellite sets because these segments are likely to have almost fixed biases.

3.2 Data Collection

3.2.1 Test Areas

In order to provide real data for analysis and experimentation several data collections were made. These all occurred in the general area of the Naval Postgraduate School in Monterey

CA (36.6 N 121.9 W). Data was collected over 3 tracks shown in Figure 7. This shows the south end of Monterey Bay, which is about 150 km south of San Francisco.

The Naval Postgraduate School is on the southern edge of this map. It is labeled NPS and is partially off the map. The static data was taken at NPS. The antenna is on top of the highest building on campus and in a multipath free environment. The reference data for the kinematic solutions was also taken at this site.

NPS has some beach property about 0.8 km from the reference site. This is called the Beach Laboratory area and marked "Beach Lab Track" on Figure 7. There is a narrow paved two-lane road on this property that was used as a test track. The road area used was about 150 m long with a large turnaround through gravel parking areas at each end. The Beach Lab area was used on several occasions over about 9 months to get repeated statistics from independent samples. Speeds were limited to about 35 km/hr (10 m/s).

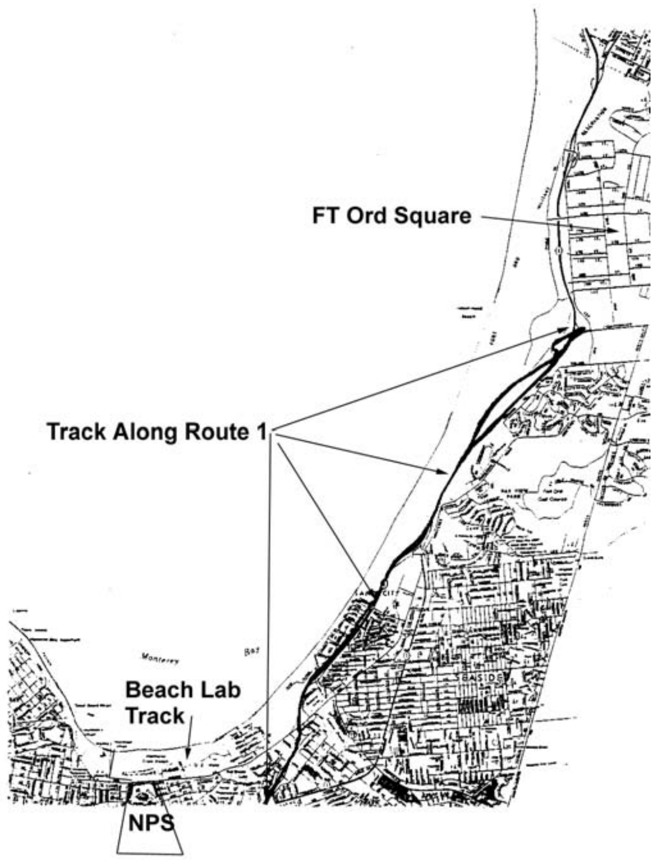

Figure 7: Three Test Areas on the Monterey Peninsula

In order to evaluate open road conditions, data was taken along California route 1 (the Pacific coast highway) over a length of about 8 km. This is a divided highway with 2 and 3 lanes, in each direction, along this area. There are no cross streets, only one underpass and no areas of limited visibility. There is limited visibility and an overpass on the cross street at the south end used for a turnaround. In all but a few controlled tests, the route took the right (slow) lane. Speeds of 100 km/hr (65 mph or 30 m/s) were common. The northern end of the route turned around at the main entrance to the old FT Ord. (This army base is now closed and converted to civilian uses.)

The straight tracks commonly found in urban areas were sampled using some streets in the former FT Ord. A rectangular route 0.7 km by 0.5 km was used in an area with little traffic. This "square" is about 10 km from the NPS reference station. It is shown in Figure 8. The visibility is good except for a few trees. In one area there are buildings that limit the horizon to about 10 degrees. Figure 8 shows the rectangle as well as the location of 4 survey markers positioned for this study (small numbers 1 to 4 inside the square). These were used in a stop and go test discussed later. It should be noted that the northern side of this route is not straight. It consists of two straight segments that join with a kink. The offset is about 25 m and occurs over a distance of about 100 m. They are also slightly offset in angle with respect to each other. This provides a nice test case for the fitting algorithms.

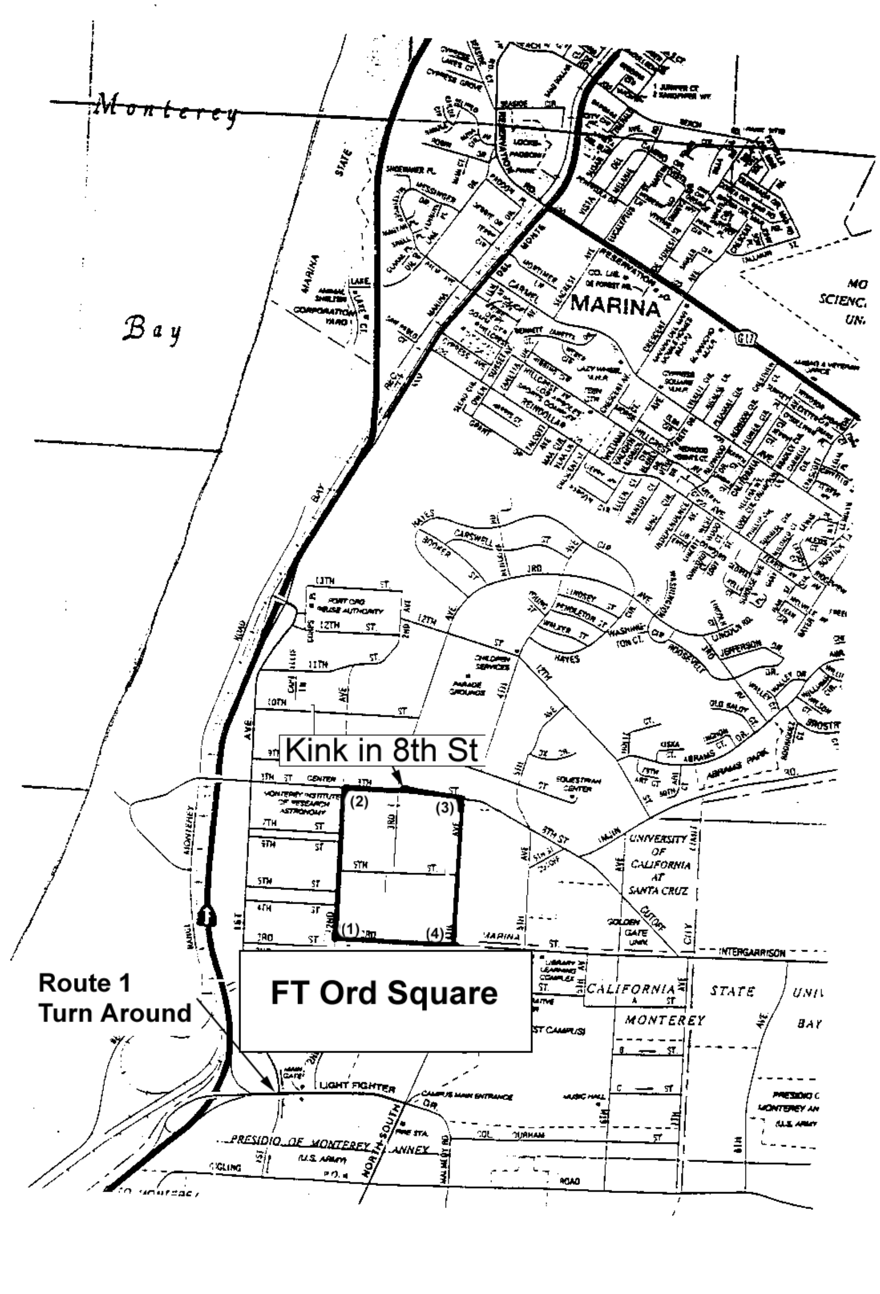

Figure 8: FT Ord Area with FT Ord Square

3.2.2 Experimental Configuration

The dynamic data was taken in all cases in a king cab truck shown in Figure 9. The receivers and data logging equipment were placed in the rear seat. An Ashtech Z12 dual frequency receiver was used to provide data for a reference trajectory. The data on this receiver was logged internally in the receiver. The reference receiver was an identical Z12 located over a surveyed mark on the NPS campus. This mark was on top of the highest building on campus in a multipath free environment. Data was taken at 1 Hz and the reference trajectory was processed with the Ashtech PNAV program.

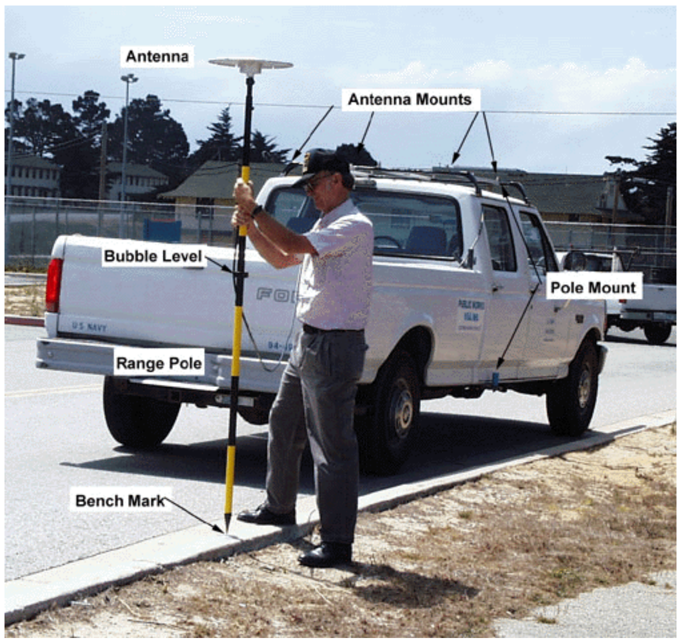

Figure 9: Vehicle Used for Data Acquisition

The three PLGR's in each test had their antennas in one of two configurations. For the first few tests they had separate antennas mounted on a square on the truck roof. The square is about 1 m on each side. The reference system was on the fourth corner. This required a lever arm correction to bring the effective location of all the receivers together. In later experiments, all the receivers shared the reference receiver geodetic antenna through a 4 way WR Inc. splitter / amplifier. This had 26 dB of gain. This common antenna was mounted on the truck roof for some runs. In others it was mounted on a pole attached to the side of the truck via a quick release. This is the configuration shown in Figure 9. This allowed the antenna and pole to be removed from the truck and placed over a survey mark. The pole had a target bubble level and a point for insertion in the survey mark.

The data from the PLGR's were collected in laptop computers using a NPS written program called VBPLOG. This program took data from the instrumentation port and converted the solutions on the fly to ASCII and logged them. (The position solutions came from PLGR data block 5040's and the velocity from block 3's.) The data were collected at 1 sec intervals.

The VBPLOG program could also control the tracking of the receivers. In all but the first test, one PLGR was left to choose its own satellites and the other two were controlled.

The tracking scenarios were generated with another NPS program. The VBPLOG program was also used to set the configuration of the PLGR to ensure that it was on the correct datum etc. The logging program also displayed the solution, DOP and tracking status. This allowed problems to be identified in the field.

In order to generate independent data sets, the data was separated into sets with different satellites being used for the solution. Only sets with two or more satellites different were considered as independent data sets if the data was taken at the same time. Data with one satellite different were ignored.

4 Static Errors

4.1 24 hour data sets, characteristic of errors

The errors of two PPS receivers, tracking the same satellites, are remarkably similar. This was dramatically observed during an at sea experiment conducted by NPS in 1996 on the Research Vessel PT SUR [8]. During that experiment there were 4 PLGR's used, two on the ship and two at a static site on shore. Each pair had only one antenna, making this a dual "zero baseline" experiment.

When the receiver solutions were differenced within each pair, the error was observed to be essentially zero over large time blocks and much larger in other blocks. It was found that the times that corresponded to very small errors occurred when the two receivers were tracking the same satellites. The tracking scenarios were available in the data, therefore statistics of the differences in bins according to the number of common satellites could be generated.

The results of this analysis for both zero baseline pairs are shown in Table 1. Here the RMS of the differences are shown for both the position and velocity. The values are in m and m/s. Cases without a significant number of points have not been listed. This causes the number in the "All Data" category to be slightly larger than the sum of the cases shown.

The cases of 4 common satellites represent the same satellites used in the solutions. Here the difference in the horizontal components is 30 cm or under on land. The vertical coordinate is about twice as large. The same pattern is shown on the ship, with about a doubling of the level.

However, when even a single satellite is different, the error jumps to the 3 m level in each component for the land case. It does not get significantly worse with a larger number of different satellites. Here the ship data is not worse, indicating that the substitution of a single satellite dominates the error budget. This demonstrates that the broadcast orbit model errors are the major error component of a PPS solution.

To illustrate this, a day of data taken in 1997 has been analyzed. In this case there was a Trimble 12 channel PPS receiver on an antenna 2 m from the PLGR antenna. The errors of both receivers as a function of time are shown in Figure 10. It is evident that the basic form of the PPS errors is the same for a solution based on the best 4 satellites and an all-in-view solution. The Trimble unit has much lower random noise, but only occasionally a much lower error value. (See the longitude error at between 08 and 10 UT.) The errors can,

A. Static Land Data

Common Sats.	Points	Δ Position (m) Lat	Lon	Height	Δ Velocity (m/s) V_n	V_e	V_u
1	881	3.08	4.00	6.83	0.011	0.007	0.031
2	9151	4.02	4.08	4.81	0.022	0.025	0.028
3	17989	3.70	3.13	4.45	0.041	0.027	0.031
4	246366	0.34	0.17	0.55	0.030	0.019	0.035
All Data	274447	1.25	1.13	1.58	0.031	0.020	0.035

B. At Sea Ship Data

Common Sats.	Points	Δ Position (m) Lat	Lon	Height	Δ Velocity (m/s) V_n	V_e	V_u
1	4	2.08	3.36	2.68	0.226	0.212	0.107
2	1702	4.04	2.82	4.15	0.191	0.128	0.115
3	11329	3.35	2.52	6.95	0.329	0.229	0.188
4	241807	0.57	0.34	0.90	0.172	0.122	0.131
All Data	254842	0.96	0.67	1.74	0.182	0.129	0.134

Table 1: Zero Baseline PLGR PPS RMS of Solution Differences By the Number of Common Satellites

however, be large in both receivers at times. See for example the height between 04 and 07 UT.

Notice that the error, for either receiver, is often the same sign for a period of 3 to 6 hours. Clearly taking shorter than a day will not significantly reduce the errors.

To further document the characteristics of the PPS error, the probability distributions of the errors were computed. These are shown in Figure 11. Here it is clear that the longitude is the best determined component. The latitude has a slightly wider and more irregular distribution. This was expected for a PLGR, but the similarity of the two in the horizontal is striking. In the vertical the PLGR is much worse. But it is a single frequency receiver. This probably accounts for the slight bias. A summary of the statistics for these data is given in Table 2.

	PLGR Avg.	PLGR σ	Trimble 12 Channel Avg.	Trimble 12 Channel σ
Latitude	0.13	3.70	0.21	1.81
Longitude	0.64	2.40	0.31	1.39
Height	-2.51	6.58	0.04	4.11

Table 2: Error Statistics for PPS Solutions Over a Day for PLGR and 12 Channel Trimble. All Values are in meters

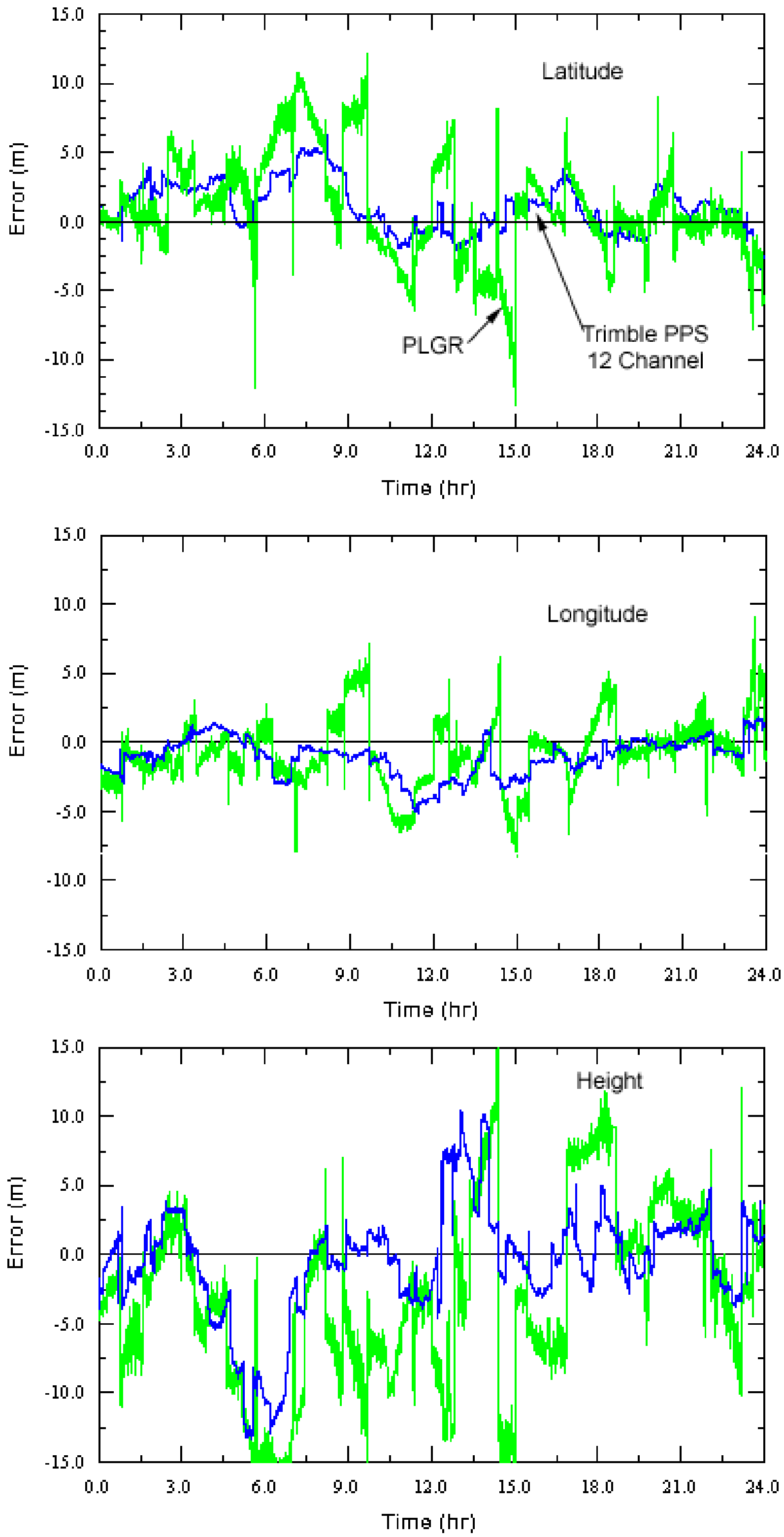

Figure 10: Errors in PPS Solutions Over a Day. A PLGR (green or lighter line) and a Trimble 12 Receiver (blue or darker line). Antennas 2 m Apart.

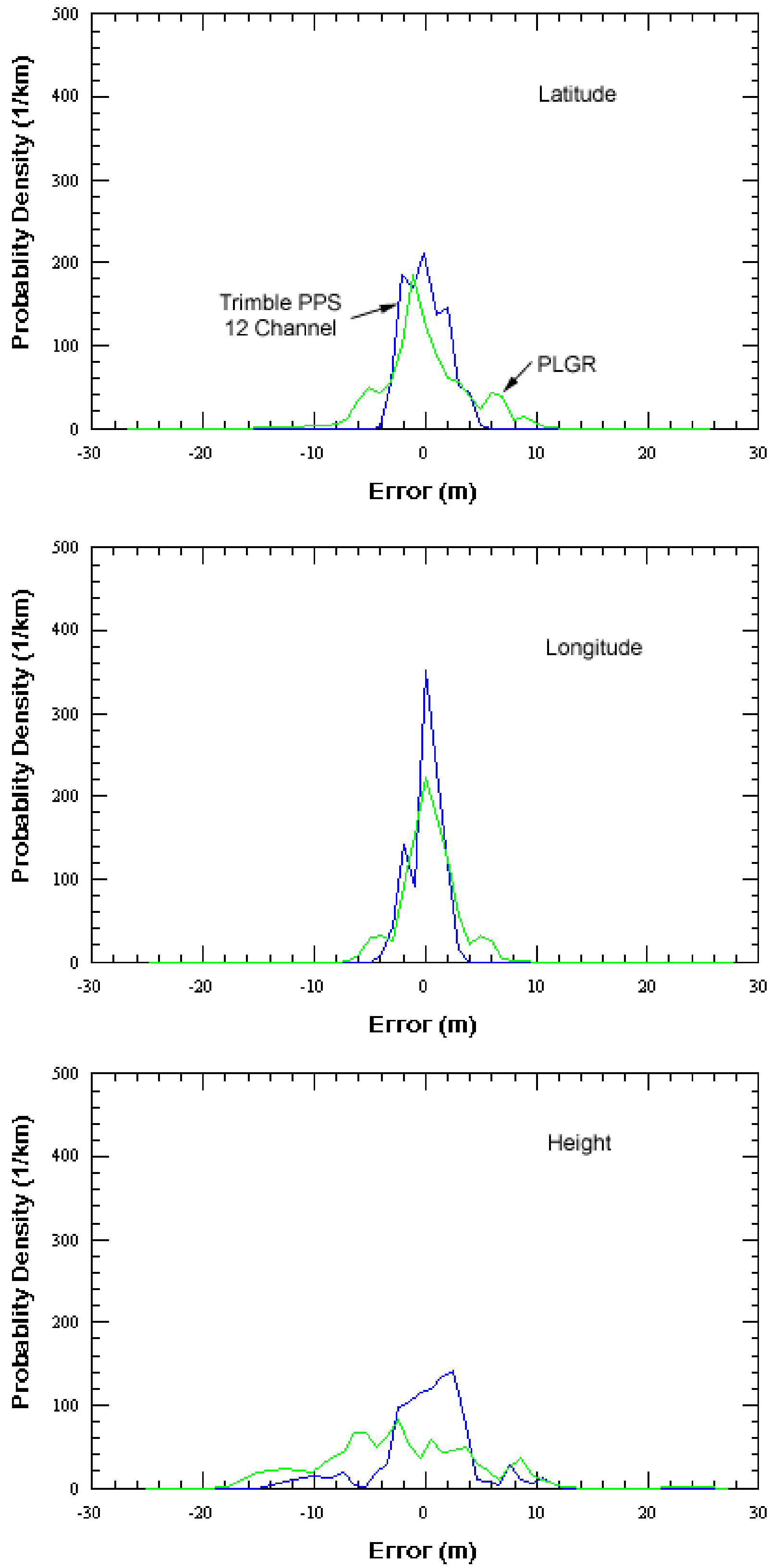

Figure 11: Probability Density Functions for Data in Previous Figure

4.2 Stop and Go

One possible technique for finding a better position at a point is to average the positions obtained in several short occupations of a point. From the previous section it is clear that the time interval between occupations needs to be large. The main requirement is that satellites change, but for a free running PLGR this often means a few hours between data sets.

In order to evaluate the validity of these assumptions, a short test was made. In this test four surveyed points were repeatedly occupied at intervals of about 10 minutes over an hour. The PLGR PPS solutions and a kinematic GPS reference solution were evaluated.

4.2.1 Experiment

Four marks were surveyed on the former FT Ord around the 0.5 km square used in this study. One marker was placed near each corner. These marks are about 10 km from NPS. A map of the area is shown in Figure 8.

A truck that had a range pole attached to its side was used. This is a straight pole about 2.5 m long with the antenna on the top and a point to insert into a survey mark at the bottom. A clamp allows quick release from the truck mount so an operator can walk the antenna to a nearby mark. (See Figure 9) Three PLGR's, NPS numbers 2, 5, and 10 and, one Ashtech Z12 were used on the truck.

For about an hour, the truck was driven around the square. At each mark, the truck pulled up just past the mark, an operator got out and set the antenna/range pole over the mark. When the pole was vertical (a bubble level is built into the range pole) he told the truck driver who recorded the time. The goal was to obtain 30 s of level data at the mark. Often more were taken. It took about 10 minutes to make a circuit. Seven circuits were made with stops. At one time a few circuits were made without stopping for other analysis.

4.2.2 Results

The data were converted to a local x-y (East, North) system for analysis. The reference point used for this conversion was a point near the Beach Lab track. The x axis was essentially a biased easting and the y axis a biased northing. Both the PPS data being evaluated and the kinematic reference solutions were treated the same.

4.2.2.1 Kinematic solution The errors in the kinematic solution can be evaluated from this data because there is a static survey on the mark. In addition the errors in the averages of the solutions while the antenna was over the mark can be obtained. These averages and the standard deviation of the data are given in Table 3. Here the errors are grouped by the mark occupied. The last column is the number of 1 second points used in each average. In general 30 to 40 seconds were taken at each site.

It is clear that the kinematic solution is very good. Only one case shows an anomaly, and this is probably due to operator problems or identifying the correct stationary data set. (There was always a stationary set with the antenna on the truck before and after each mark observation.) The errors are generally in the 1 to 2 cm level. This is extremely good for a solution that is advertised to be good at the 5 to 10 cm level.

	Error			Standard Deviation			
Mark	East	North	Up	East	North	Up	Npt
1	0.00	-0.01	0.03	0.02	0.01	0.00	38
1	0.00	0.00	0.02	0.01	0.00	0.01	44
1	0.38	-0.32	-0.92	0.20	0.19	0.50	37
1	0.03	-0.03	-0.02	0.06	0.07	0.03	36
1	0.00	0.00	0.00	0.00	0.01	0.01	32
2	0.00	-0.01	0.00	0.01	0.01	0.00	147
2	0.01	-0.02	-0.03	0.00	0.00	0.00	62
2	0.01	-0.01	-0.02	0.01	0.01	0.01	117
2	0.02	-0.02	-0.02	0.00	0.00	0.01	38
2	0.00	-0.01	-0.03	0.01	0.01	0.01	39
2	0.01	-0.01	-0.02	0.00	0.01	0.00	42
3	0.00	-0.01	0.00	0.00	0.01	0.00	40
3	0.00	0.01	0.01	0.01	0.01	0.00	44
3	0.00	0.00	0.00	0.01	0.01	0.00	38
3	0.00	0.00	-0.01	0.01	0.01	0.00	29
3	-0.01	0.02	0.02	0.00	0.01	0.01	30
3	0.01	0.00	-0.02	0.01	0.01	0.00	37
4	0.00	-0.01	0.01	0.01	0.01	0.01	38
4	0.00	0.00	0.03	0.00	0.01	0.00	34
4	0.01	0.01	0.01	0.01	0.01	0.01	36
4	0.00	0.01	0.01	0.01	0.01	0.01	30
4	0.00	0.00	-0.01	0.01	0.02	0.00	25

Table 3: Kinematic Reference Solution Errors. All values are in meters.

4.2.2.2 PLGR PPS Absolute Positions A similar analysis was done on the PLGR solutions. In this case the data were first separated by receiver and then by the location. There is a table for the error of each receiver. These are given as Tables 4 - 6. In these Tables, a scenario number is also listed. This is because the satellites being tracked are much more important than the receiver being used. The satellites tracked in each scenario are given in Table 7.

The horizontal errors from scenarios 1 and 2 are shown in Figure 12. The same plot for all the data is given in Figure 13. The standard deviations of the data in the set are plotted as error bars. It is very clear that the internal consistency of the data as seen in the standard deviations is usually much smaller than the true errors. It is also clear that the “bias” is slowly walking.

There is a significant difference in the standard deviations of the data in the two major scenarios. In part this is due to the higher DOP for scenario 2. For scenario 1 the DOP is in the range 2.5 to 4 while for scenario 2 the range is 2.9 to 6. Other factors may also be

	Error			Standard Deviation				
Mark	East	North	Up	East	North	Up	Scn	Npt
1	3.20	0.61	0.46	0.12	0.68	0.39	1	36
1	4.43	1.94	0.23	0.14	0.13	0.34	1	42
1	4.27	2.95	0.72	0.26	0.20	0.43	1	36
1	4.76	4.35	2.30	0.31	0.33	0.42	1	25
1	-0.76	-2.32	-9.12	0.57	0.19	1.42	7	27
2	2.91	0.51	0.35	0.38	0.67	0.79	1	144
2	3.00	0.64	0.56	0.22	0.23	0.38	1	59
2	3.86	1.94	1.24	0.17	0.41	0.55	1	114
2	4.72	2.53	-0.56	0.13	0.31	0.25	1	39
2	5.40	4.63	2.65	0.31	0.20	0.86	1	34
3	2.52	0.50	-0.67	0.10	0.42	0.41	1	41
3	3.90	1.45	0.83	0.20	0.20	0.52	1	39
3	4.88	2.55	0.53	0.15	0.23	0.86	1	37
3	5.17	2.81	-0.69	0.10	0.22	0.40	1	26
3	-0.50	-1.99	-8.15	2.34	1.25	4.18	3	28
4	3.07	0.60	0.03	0.23	0.17	0.10	1	34
4	3.44	1.64	1.04	0.20	0.31	0.38	1	32
4	4.11	2.00	2.47	0.14	0.14	0.35	1	34
4	4.70	3.38	-1.35	0.27	0.24	0.64	1	10
4	4.43	3.00	-2.47	0.07	0.10	0.14	1	14

Table 4: PPS Errors for PLGR 2 at Survey Markers. All values are in meters.

at work here. The very large error bars in the "one of" cases may be influenced by recent satellite changes that have not yet caused the solution to stabilize at a new bias.

	Error			Standard Deviation				
Mark	East	North	Up	East	North	Up	Scn	Npt
1	3.29	0.46	0.57	0.81	0.56	0.51	1	24
1	4.33	1.73	0.35	0.13	0.06	0.22	1	12
1	4.37	2.74	0.80	0.17	0.12	0.48	1	20
1	5.00	4.39	2.98	0.10	0.09	0.36	1	12
1	4.66	5.41	3.77	0.08	0.23	0.57	1	11
2	-0.26	0.57	4.26	3.98	3.69	2.33	8	12
2	2.89	0.38	0.60	0.09	0.25	0.43	1	26
2	2.82	0.77	0.65	0.14	0.13	0.35	1	10
2	4.05	1.86	1.27	0.10	0.17	0.15	1	28
2	3.49	1.90	1.41	0.09	0.33	0.38	1	27
2	4.55	2.31	-0.53	0.10	0.15	0.20	1	24
2	5.35	4.43	2.68	0.22	0.09	0.85	1	33
3	3.61	1.39	1.48	0.08	0.11	0.13	1	11
3	3.92	1.26	0.49	0.17	0.09	0.40	1	26
3	4.87	2.40	0.26	0.14	0.16	0.67	1	21
3	5.05	2.84	-0.03	0.09	0.13	0.24	1	16
3	5.92	4.54	-0.38	0.12	0.14	0.57	1	10
3	5.88	4.76	2.86	0.19	0.12	1.38	1	13
3	5.75	5.80	7.90	0.05	0.16	0.42	1	18
4	3.40	1.49	1.19	0.21	0.27	0.36	1	26
4	4.12	1.84	2.58	0.10	0.10	0.34	1	16
4	4.50	2.93	-2.12	0.27	0.26	0.71	1	13

Table 5: PPS Errors for PLGR 5 at Survey Markers. All values are in meters.

Mark	Error East	Error North	Error Up	Standard Deviation East	Standard Deviation North	Standard Deviation Up	Scn	Npt
1	2.78	-2.68	-1.32	0.54	0.57	0.26	2	36
1	3.92	-3.46	-2.58	0.16	0.14	0.46	2	28
1	2.65	-4.99	-0.41	0.12	0.50	0.69	2	26
1	3.51	-4.54	-5.31	0.29	0.23	0.40	4	34
1	2.81	8.49	-5.63	0.34	0.04	0.74	5	10
2	2.60	-1.70	-1.24	0.36	0.57	0.78	2	128
2	2.56	-2.84	-1.20	0.85	0.20	0.37	2	56
2	3.38	-3.89	-1.64	2.40	0.46	0.93	2	95
2	1.99	-5.90	-1.08	0.52	0.23	1.50	6	36
3	2.15	-2.32	-2.48	0.31	0.38	0.44	2	41
3	3.38	-3.38	-1.78	0.09	0.20	0.41	2	37
3	4.57	-4.21	-3.37	0.37	0.11	0.44	2	17
4	2.61	-2.35	-1.72	0.19	0.12	0.18	2	36
4	2.94	-3.69	-1.61	0.39	0.25	0.28	2	32

Table 6: PPS Errors for PLGR 10 at Survey Markers. All values are in meters.

Scenario	Satellites Prn's				Number Stops
1	1	8	15	25	20
2	1	8	15	3	17
3	14	8	15	25	1
4	29	8	15	3	1
5	29	14	15	21	1
6	29	23	15	3	1
7	25	14	15	21	1

Table 7: Track Scenarios For PLGRs

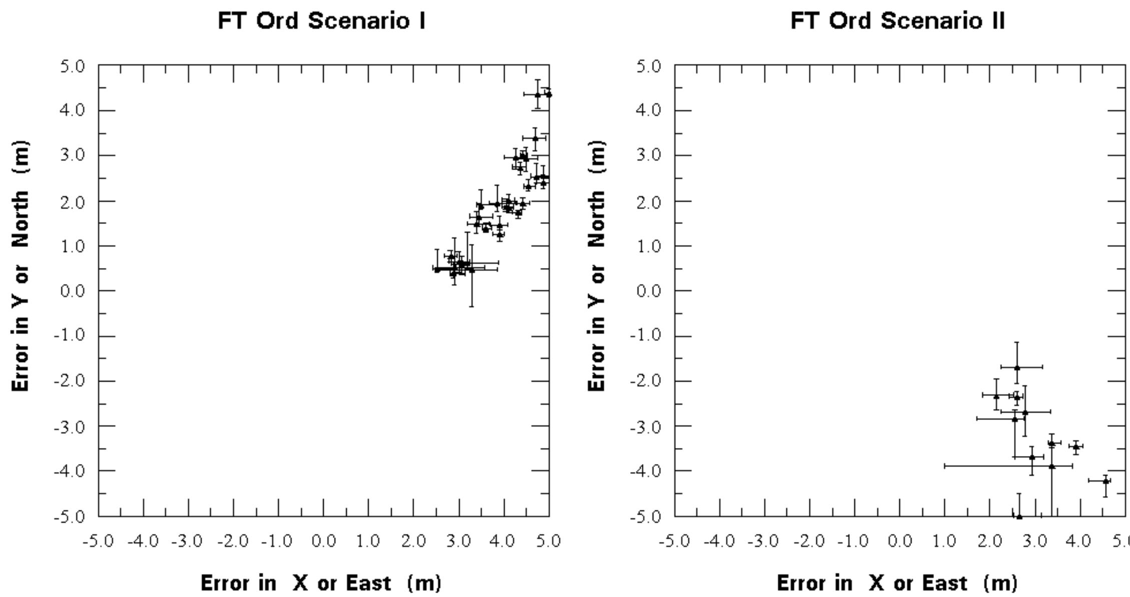

Figure 12: Errors at FT Ord Stops (a) Scenario 1 (b) Scenario 2

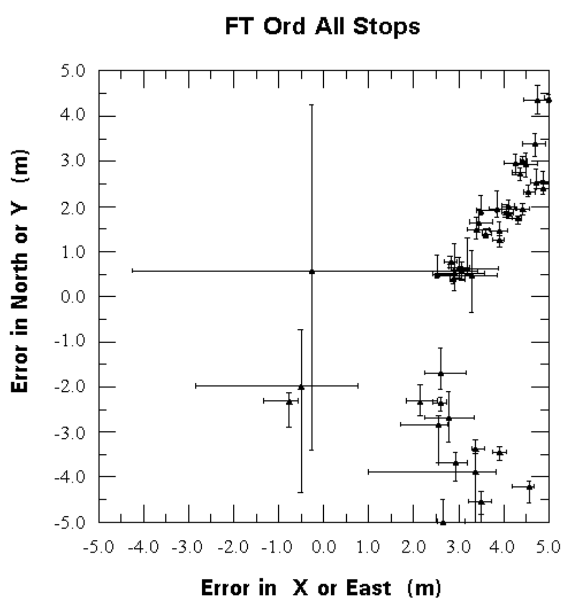

Figure 13: Errors at FT Ord Stops, All Data

4.2.3 Stop and Go Summary

The noise level due to the inherent variation in a PLGR solution is at the 0.2 m level in most cases. There may be some receiver to receiver variation. This is for a DOP of 3.

The "biases" walk. The typical velocities are 5 m / hour. Therefore one should not use segments of data longer than about 10 minutes in a system trying to define positions at the 1 m level.

5 Dynamic Approach

5.1 Model Assumptions

In the analysis of data from PPS GPS receivers it will be assumed that the Clock and Orbit errors inherent in the use of the broadcast ephemeris dominate the error. This means that for the present analysis, we are ignoring environmental effects such as multipath. It will also be assumed that the random noise contribution is much smaller than the Clock and Orbit errors.

In particular it is assumed that the error in a position will have two major components:

1. A small random component, here assumed to be about 25 cm per axis in the horizontal plane,

2. A larger error that changes only slowly while a fixed set of satellites is used in the solution. (In reality the assumption is that a fixed set of satellites with broadcast ephemeris from the same upload. Within that upload, epochs or IODE/IODC's can change.)

   This larger error:

   (a) Can be modeled as a constant or linear function of time. Over a time scale of 10 to 15 minutes it can be considered a constant.

   (b) Will change discontinuously when satellites used in the solution change.

These data will be converted to space tracks, removing the time as an independent variable. It is assumed that space tracks over the same short segment of road will have an error that is a bias with respect to the "truth". It will be assumed that these bias vectors are independent for different satellite sets or on different uploads. It is assumed that the error in these bias vectors is random and has a zero mean.

5.2 Mathematical Overview

5.2.1 Tracks from Biases

Let the true track segment be $T(s)$, where s is some measure of the distance along the track. There will be n sets of measured locations over this same physical track segment. Based on the assumptions, these will be the true track segment plus a bias vector plus some random component.

The first step will be to take the discrete, time ordered, GPS locations and fit them to an analytic curve in space. One benefit of this process is to average out the random component. Also some of the driving errors will be removed. We will denote the fit to a measured track segment by $T_i(s),\ i = 0, 1, \ldots, n-1$. Then the basic assumption is made that

$$T(s) = T_i(s) + \beta_i,$$

for all n track segments.

In the real world, the true track segment is unknown and only $T_i(s)$ are available. The approach is to choose one track segment as a reference track. Here track segment zero will be chosen. The offset between each of the track segments and track segment zero will then be estimated,

$$\begin{aligned} \Delta_i &= < T_i - T_0 >_s \\ &= \beta_i - \beta_0 \end{aligned}$$

Here $< ... >_s$ denotes the average over the distance measure s.

Now the average of the Δs over track segments will be taken

$$\begin{aligned} < \Delta_i >_i &= < \beta_i >_i - \beta_0 \\ &\approx -\beta_0 \end{aligned}$$

Of course this average does not include the reference track segment because Δ_0 is always identically zero. Here it is assumed that the bias vectors are random and will average to zero given a sufficient number of samples. Thus

$$\begin{aligned} T &= T_0 + \beta_0 \\ &\approx T_0 - << T_i - T_0 >_s >_i \end{aligned}$$

The average over the track segments can be done as a simple average. However it is more appropriate to do a weighted average using some measure of track quality. Two estimates have been studied here. The first is the post-fit rms from the offset vector solution process. A second method is to use the N-Corner Hat method of Barnes [9] popularized in the precise timing community by Allan [10]. This method takes the above rms values from solutions between all pairs of tracks segments and estimates the most likely variance of each bias vector. In both cases the reciprocal of the variance or rms squared is used as the weight.

In the cases studied here the track segmens are vectors in two dimensions and the β's are two-dimensional vectors. It is important to note that the β, and hence the Δ can only be estimated if there is significanmt variation of the track in the two components of the segment studied. If the track segment is straight, only the cross track component of the Δ's can be resolved. This will manifest itself in a singular covariance matrix between two track segments. In this case a solution for only the cross track component of the offset vector will be found.

An example using nine independent track segments following the same path will be given in Section 7.1. It is important to mention that for a straight line, the solution for β is singular, one can only find cross track coordinate, not along track component. This is why we discuss 1-d fit in section 6.4.

5.2.2 N-Cornered Hat Test and Variance Calculations

The N-cornered hat calculation was designed to estimate the variance in a sequence of time estimates of N independent clocks [9]. The basic equations are obtained in the following way. Let T^i represent the time sequence from the i^{th} clock, with unknown variance σ_i^2, and T the true time sequence. The the matrix of variances of the differences between the observed sequences can be computed

$$\begin{aligned} S_{ij} &= var(T^i - T^j) \\ &= var(T - T^i) + var(T - T^j) \\ &= \sigma_i^2 + \sigma_j^2 \end{aligned}$$

can be computed. The function "var" is the variance of its argument. Here it is assumed that the sequences are zero mean and uncorrelated. This relates the computable quantity, S_{ij}, to the variances of the individual clocks. We then have S_{ij} for $i = 1, \ldots, N$, $j = i+1, \ldots, N$, providing $\dfrac{N(N-1)}{2}$ equations in the N unknown variances. If $N > 2$ there are at least as many equations as unknowns, and the approximate value of the variances can be found by least squares methods. For ease in writing the equations, assume that $S_{ji} = S_{ij}$ for all ij with $S_{ii} = 0$. The least squares estimate results in the solution

$$\sigma_i^2 = \frac{1}{N-2}\left(\sum_{j=1}^{N} S_{ij} - \frac{1}{2(N-1)}\sum_{k=1}^{N}\sum_{j=1}^{N} S_{kj}\right), \; i = 1, \ldots, N.$$

This calculation may result in negative variances under certain conditions, and that is observed to occur when the true variance of the clocks is significantly larger than that of the others. In that case the calculation can be used to determine a clock with a large variance, eliminate it from the set and repeat the calculation.

We have used this procedure in a slightly different setting. When the bias calculation is done (see section 7), the mean-squared-error from the calculation of the offset vector between two curves replaces the variance calculation above. We are then able to estimate the variance of the error between the true track segment and the given test track segment. When we performed this calculation for the nine track segments, it was found that the variance for one was relatively large while the variance for another was negative. This unphysical result was corrected by removing the track segment with the very large variance from the set and the calculation repeated. This gave good estimates of the variance of each track segments' errors.

5.2.3 Generating 3 Dimensional Space Tracks

For a single track of data, two methods of fitting the data in 3-d seem apparent. The first, and most difficult, is to extend the Bezier cubic fits, discussed earlier, to 3-d. Knot points would have three components, the tangents at the knot points would have two degrees of freedom, while the distances would be the same (two per cubic segment). This is relatively straightforward to implement and results in $7 * k - 2$ parameters for a k knot Bezier cubic. Of course, the errors in the z-component would be weighted differently than those in the x and y-components.

A second, and easier method is a two step procedure. Fit the $x - y$ data first. The parameter value for each point is then available (or easily computed). The distance along the curve could also be easily computed. The z-component could then be fit as a function of either parameter value or distance along the curve (it's suggested the latter is a better idea) using the 1-d analogue of Bezier curves, Bessel cubics. Since the z-component has much larger error than the horizontal component, this approach seems attractive because it decouples the problem into two simpler problems.

If a single path in horizontal coordinates is generated through "averaging" the data from several paths, the method of then estimating the height along the resulting curve from the z-component data is not so clear-cut. The problem is attempting to identify a parameter (or distance) value of each point with the z-value. Since different paths have different biases, this could only be done by taking into account the bias between the "averaged" curve and the individual curve that the z-component datum came from. This could be done, but it is not clear that the z-data should be treated this differently.

Instead, it seems reasonable to "average" z-components from several paths using an algorithm similar to that used for the horizontal coordinates.

6 Dynamic Space Tracks

6.1 Considerations - linear/2-d, point spacing

We have developed Matlab codes for data segmentation and track averaging. In this section, we discuss the data segmentation to pick independent tracks and to choose pieces that should be fit by a straight line (see section 6.4). The latter is required since in this case one can only find the cross track error.

6.2 Data segmentation

For the purpose of this study the data segmentation was done in a semi-automated fashion. One program finds the segments of tracks which are monotone in "x". It also finds times at which satellite groups change. Plots are made of each segment, along with a timeline plot of the various path segments and satellite groups.

The program then interrogates the user for a time interval or segment to be "picked off". One or more segments are then saved in .mat files specified by the user (the name is the same as the input file with an index to distinguish between the segments). Some of these may then be further reduced to track segment data sets that can be fit by piecewise cubics or a straight line.

6.3 2-d fit mathematics goodness of fit (rms of fit vs. rms errors)

The initial guess for the knot locations is given to the fitting program graphically. The data is displayed with labels indicating the order. Using the mouse, the user indicates the desired location of the knots for the cubic pieces. All data before the data point closest to the first knot and after the data point closest to the last knot is discarded. Kept and discarded points are indicated on the graph and the user is given the option of accepting the input, or restarting the knot selection process. The approximating curve is then computed. Graphical output is supplied. This data is then saved.

The placement of and number of knot points plays a crucial role in how well the initial curve and ultimately how well the optimized curve fits the data. Experience is the best teacher of how to do this, but there are some hints that can be given. Recall that the curve starts at one end (knot) and ends at the other (the second knot), and is tangent to the corresponding polygonal segments. In between there are two control points, the vertices of the polygon segments whose placement is determined by the program. The Bezier curve will rarely pass through either of these control points.

The initial guess algorithm is dependent on an ordering of the input points, and is taken as the input order, with every tenth point annotated. The user then indicates (with the same orientation) a set of knot points for the initial guess, using the mouse to place a cursor. All points preceding the first indicated knot, and subsequent to the last indicated knot point are discarded from the data set.

The shape of the data curve will determine the number of knot points required for the complete curve. While it is possible to fit data with an inflection point in the interior of a single parametric cubic segment, it is probably a good idea to insert a knot point at the approximate location of the inflection point. Other knot points should be inserted commensurate with the shapes that are possibly generated by a single parametric cubic curve.

Generally, it is felt to be a good idea to use no more than 3 or 4 cubic segments (4 or 5 knots). If suitably small errors are not obtained in a particular case, it is necessary either to increase the number of knots, or to decrease the extent of the data being fit. As the present time, no software for automatic placement of additional knots, nor refinement of them after an unsuccessful approximation is available.

With a little experience the user can select segments of the data and supply initial guesses that result in the approximation having rms errors (of the distance of the data points from the fitting curve) that are on the order of 0.5 meter and sometimes less. Such errors are in line with the errors shown in Figure 3 for the "random" component and excluding the larger bias errors that appear to be approximately linear in time. For a mathematical discussion of these errors, see Section 5. For a short time interval the fitted curve is primarily in error due to the bias error since the random error is greatly diminished by the curve fitting process.

6.4 1-d fit mathematics goodness of fit

When data is collected along a straight road, it is desirable to fit this data using a straight-line segment. This is accomplished using a "total least squares" fit by a straight line. This process determines the coefficients in the approximation by minimizing the distance from the data points to the line. Our algorithm attempts to find significant segments of essentially linear data collection by sequentially fitting subsets of the data using this process. If the rms error of the fit is greater than a specified value, the algorithm decreases the amount of data considered, and attempts the process again. If less than a specified number of points remain, it is assumed the data was not collected from an approximately straight-line segment. By using the rms tolerance of the fit, one can find straight-line segments with error that is

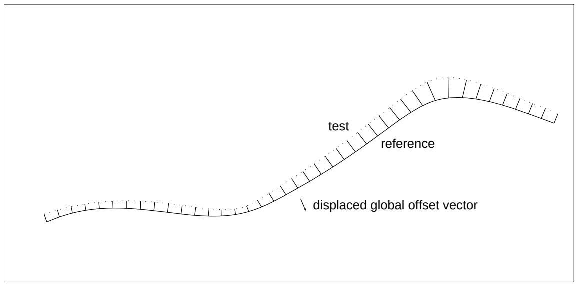

Figure 14: Local Offset Vectors

commensurate with the random error in the data, leaving the bias error as in the case of curve fits. The straight-line data can be converted to the more general curve form if this is desirable.

7 Track Averaging

The first step in track averaging is to estimate the bias between two curves that represent (approximately) the same track segment. It is assumed that there is a current estimate of the track segment, represented by a curve, here called the "reference" curve. The second curve will be called the "test" curve. First, a set of equally spaced points is generated on the reference curve. For each of these, the closest point on the test curve is found, and the vectors resulting from joining the corresponding points, from reference curve to test curve are found (see Figure 14). Call the vectors from the test curve to the reference curve local offset vectors. We now find a fixed vector (the global offset vector) so that the length of the projection of the global offset vector onto the local offset vectors is equal to the length of the local offset vector. This is an overdetermined problem, and the solution is by least squares, yielding the single offset vector from the reference curve to the test curve. This vector would be the negative of the bias vector if the reference curve is considered to be accurate. The standard deviations and the correlations between the errors in the two components are also computed. An example of the two curves and every fifth local offset vector is shown in Figure 14, along with the (displaced) computed global offset vector.

7.1 Results

We have available nine statistically independent runs (at least two satellites different) on the eastbound portion of the beach lab road. In addition, we have the "truth", a high resolution set of data for one of the track segments. Using this data we computed the offset vectors for each of the nine data sets relative to the "truth" data.

Figure 15 (left) shows the nine track segments (the eight runs plus the reference track). The right figure shows the eight test tracks translated by the relative offset vectors(Δ's) to

Run	Real Errors		Run 0 Differences		Fit RMS	N-Cornered-hat
	East	North	East	North		σ_i
0	1.39	-0.57	0.00	0.00	0.25	.31
1	3.45	-2.95	-2.06	2.38	0.38	.30
2	-0.93	0.30	2.32	-0.87	0.55	.46
3	0.48	1.83	0.91	-2.40	0.57	.39
4	-1.64	5.59	3.03	-6.16	2.50	omitted
5	-0.29	2.85	1.68	-3.42	0.46	.50
6	0.27	1.44	1.12	-2.01	0.60	.47
7	0.53	5.66	0.86	-6.23	0.43	.25
8	-0.62	-2.43	2.01	1.87	0.30	.37
Mean	0.29	1.30	1.23	-2.11		
σ	1.40	2.92	1.43	3.02		

Table 8: Error Vectors and Relative Offset Vectors for 9 Track segments at Beach Lab Test Area

be aligned with track segment 0. The offset vectors, and the true error vectors (β's) for these 9 track segments are listed in Table 8. With the exception of one track over part of the curve, the set is very consistent considering the data were taken by driving the path nine different times. In addition this Table also lists the post fit root-mean-square error and the estimate of the standard deviation obtained from the N-cornered-hat procedure.

The estimates of the variances from this N-cornered-hat computation are given in Table 9. In the first estimate using all 9 runs the variance of run 4 was very large and there is one negative variance. Clearly a negative variance is not meaningful. This is caused by the very large value of run 4. When that run is omitted the values are all positive and reasonable. For comparison the mean square of the errors in each track are also listed.

	σ_i^2		Mean Square
Run	All Runs	Omitting 4	Error vs. Truth
0	0.224	0.099	0.063
1	0.190	0.090	0.145
2	0.115	0.213	0.293
3	0.005	0.155	0.329
4	5.686		6.252
5	0.308	0.252	0.211
6	0.089	0.224	0.362
7	-0.019	0.063	0.184
8	0.321	0.137	0.092

Table 9: Vatriance Estimates from N-Cornered-Hat Procedure. All values are in m^2

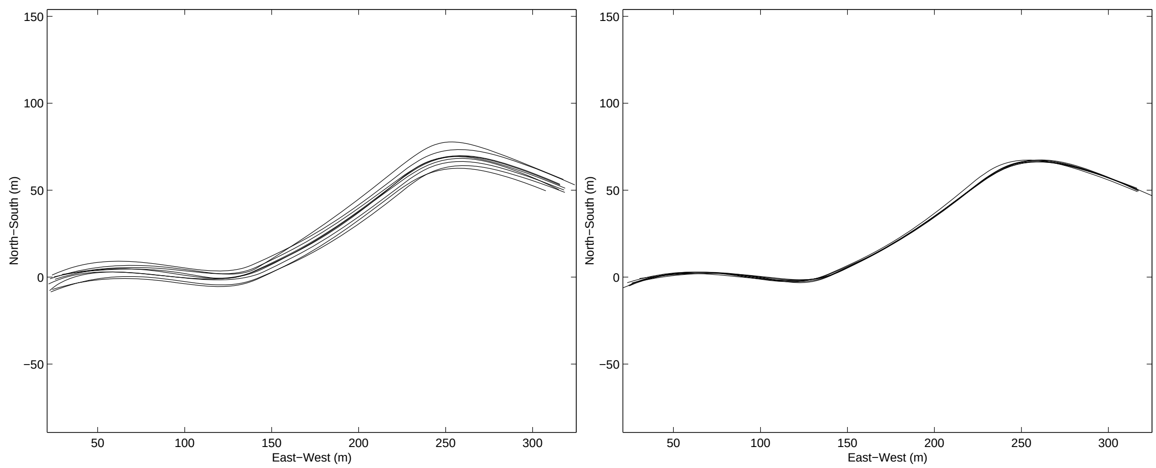

Figure 15: Nine Independent Tracks over Beach Lab Track. (a) Raw PPS Solutions, (b) After Removal of Intertrack Biases

Their average error was now computed five ways. First it was computed without weights using all 9 tracks and then omitting track 4, the one that does not appear to be a member of the ensemble. Then the same computation was done using the rms of fit in the weighting. Finally the estimates of standard deviation from the N-cornered-hat procedure were used. In this case only the data set omitting track 4 was used. In each case the weights were one over the variance (or 1 for the unweighted cases). The results are shown in Table 10.

Data Set	Weight Type	East		North	
		Avg	σ	Avg	σ
All Data	Unweighted	0.29	1.40	1.30	2.92
Omit Run 4	Unweighted	0.54	1.30	0.77	2.64
All Data	Fit RMS	0.74	0.59	-0.02	1.23
Omit Run 4	Fit RMS	0.75	0.55	-0.03	1.12
Omit Run 4	N-Cornered Hat	0.86	0.51	1.02	1.04

Table 10: Average Offset Vector for Different Weights and Data Sets. All values are in meters

We expect the average offset of the test track segments from the true track segment, given in the left two columns of Table 8, to be approximately (0,0). The average value of the offset vector is (0.29, 1.30). Because the sample size is nine, the standard deviation of the average (1.40, 2.92) is decreased by a factor of $\sqrt{9}$ to get (0.46, 0.97) to get an estimate of the uncertainty in the average. In addition, the results from the more sophisticated procedures are listed in Table 10. In all cases, the average error is under a meter. The formal errors give a good idea of the size of the error, although they overestimate the accuracy a little. Part of this may be due to driving errors.

There are three places where estimates of errors come into this process. The first is the accuracy of fitting the raw positions to the space curves. That process has an error estimate of 0.4 m. The second is the fitting of the Δ's. This process is dependent on the geometry of the track and especially if there is variation in both directions. Here the variation was mainly in the east-west direction, meaning that the east-west component was less well determined than the north-south one. In fact the covariance matrices from that process predicted the error to be about 2.5 times as large in the east-west direction. However, when we examined the variations of the average Δ's, the east-west component has about half the scatter as the north-south. This must be due to an inherent bias in the PPS positions at mid-latitude.

7.2 Convergence

Using the biases computed from the true curve, a test was administered to the coordinates (individually) of the biases to determine whether they are consistent with the hypothesis that they are from a normal distribution. Because there are only nine points, a Chi Squared Test cannot be administered. It was decided to use a variation of the Kolmogorov-Smirov Test called the Lilliefors Test [11].

The null hypothesis is that the sample is from a normal distribution with unspecified mean and variance. The test compares an empirical cumulative distribution having zero mean and variance one that is derived from the data, with a normal cumulative distribution with mean zero and variance one. The test statistic is the maximum difference between the empirical and normal cdfs, and a table determines whether the test rejects or accepts the hypothesis at a given level of significance.

For the given data, the test statistic yields acceptance of the hypothesis at all levels of significance below about 25%. This holds for the components of the bias in the two directions, independently. This means the sample of biases are consistent with being from a normal distribution. The alternative conclusion would result in rejecting more than 25% of samples from a normal distribution.

Thus this limited data set is consistent with the results converging to the true track as a normal distribution. Therefore convergence as $1/\sqrt{N}$ is expected.

8 Summary and Conclusions

The assumption that the errors in the broadcast message dominates the error in a Precise Positioning System GPS system has been investigated. Tests in both static and dynamic conditions were carried out. A method of adjusting dynamic tracks to allow their averaging was demonstrated.

The major conclusions of this study follow. The first few are essentially the assumptions that were made going into the study, which have now been validated with experimental data. The latter conclusions come from a particular implementation of "track averaging".

1. The error in PPS solutions is a slowly varying function of time given the same satellites are used with the same broadcast ephemeris. These errors are dominated by the broadcast ephemeris errors.

2. If the set of satellites or ephemeris changes there is a step change in this error.

3. Given the same satellites and ephemeris, the error can be treated as a bias vector over periods of 10 to 15 minutes at the 1 m level.

4. "Biases" in measurements with two different satellites/ephemeris in a 4 channel receiver can be treated as independent measurements.

5. The tracks of a road measured multiple times with PPS receivers can be averaged through the use of "space curves". These are functions of the position parameterized based on the spatial variation rather than based on the times of observation. Curves that fit the data to 0.4 m were easily achieved.

6. A piecewise Bezier parameterization is well suited to represent these space curves. It can fit road data to under 0.5 m with an economy of parameters. It can easily accommodate corners and sharp curves as well as straight segments.

7. Solutions for the biases between different tracks in the horizontal can resolve two parameters if the tracks vary in two dimensions. If the track is essentially linear, only the cross track difference is resolvable.

8. An example of 9 tracks was found to have statistically random bias vectors.

9. In an implementation based on Bezier space curves, a small road segment was fit to under a meter with 9 measurements. A method identifying tracks poorly fitting the ensemble was demonstrated for this case.

References

[1] Malys, S. et al, “The GPS Accuracy Improvement Initiative”, Proc. ION GPS-97, Kansas City, MO, September 16-19, 1997, p. 375.

[2] Ahlberg, J. H., Nilson, E., and Walsh, J., “The Theory of Splines and Their Applications”, Academic Press, 1967.

[3] Foley T., and Nielson, G. M., “Knot Selection for Parametric Spline Interpolation”, in T. Lyche and L. L. Schumaker, eds., Mathematical Methods in Computer Aided Geometric Design, Academic Press, 1989, p. 261.

[4] Marin, S. P., and Smith, P. W., “Parametric approximation of data using ODR splines”, Computer Aided Geometric Design, Vol. 11, 1994, pp. 247–267.

[5] Farin, G., “Curves and Surfaces for Computer Aided Geometric Design”, Academic Press, 4th edition, 1997.

[6] Holmes, M. R., “Least Squares Approximation by G^1 Piecewise Parametric Curves”, M.S. Thesis, Naval Postgraduate School, Monterey, CA, 1993.

[7] Lane, E., “Fitting Data Using Piecewise G^1 Cubic Bezier Curves”, M.S. Thesis, Naval Postgraduate School, Monterey, CA, 1995.

[8] Clynch, J.R., “One Meter Positioning on a Ship with PLGR”, Proc. ION GPS-97, September 16-19, 1997, Kansas City, MO, 973.

[9] Barnes, J. A., “Notes from the NBS 2^{nd} Seminar on Atomic Time Scale Algorithms”, June 1982, Nat. Bur. Standards, Boulder, CO.

[10] Allan, D. W., “Time and Frequency (Time Domain) Characterization, Estimation, and Prediction of Precision Clocks and Oscillators”, IEEE Trans. UFFC, Vol. 6, p. 647.

[11] Conover, W. J., “Practical Nonparametric Statistics”, John Wiley and Sons, 1980, p. 357.

Appendix A

The following functions were also developed for the project.

bezcub.m	evaluates Bezier cubic at input parameter values.
cpplt.m	plots curve and points, given input parameter array and (optional) points.
ctrpts.m	calculates control points from knots, angles, and distances.
distpts.m	calculates the rms of distances from input data points to curve described by array of parameters.
espovi1.m	takes as input the data from two fits. It computes quasi-equally-spaced points on the reference curve. Then the nearest points to each of these on the test curve are computed. Finally, ofsetvec is called to compute the vector offset between the two curves along with the variance and the correlation matrix for the fit.
gpsfit2.m	main driver for Least Squares Approximation By G^1 Piecewise Parametric Cubics for the gps data.
iktgd.m	forms curve description vector from knots, angles, distances
initg.m	computes the initial curve description array from initial knot points and data.
knotgues.m	interrogates user for initial guess knot points, given data. Called by gpsfit2.
ktgd.m	extracts knot points, angles, and distances from curve description array.
localopt.m	computes sum of distances from points to curve for local optimization. Called by fmins.
mat2asc.m	extracts from a .mat file the information about the data points, the fit, and the closest points and outputs this as an ascii file.
objdist.m	computes sum of squares of distances from data points to given Bezier curve, plus penalties for start and end. Called by fmins.
offcurv.m	computes distance between two curves as a function of distance along the reference curve, and plots it.

Table 11: Matlab Functions and Their Purpose

offsetv.m	computes offset vector between two curves - called by espovi and espovi1.
optseg.m	computes optimized distances for single Bezier cubic. Called by gpsfit2, and uses localopt.
plcpsp.m	plots control points and knots of Bezier curve from input description array.
plhash.m	plots hash marks on Bezier curve to separate cubic segments.
psegsat.m	plot the trip segments and satellite groups for arbitrary input data files.
refcom16.m	compares input data with reference (truth) data for July 16.
sepl.m	takes the output of sepseg.m and finds a piece of that data which is linear using total least squares
tlsl.m	computes total least squares fit line for input data.

Table 12: Matlab Functions and Their Purpose - continued

Appendix B: Optimization routines

The program **gpsfit2** uses **fmins.m** first in the local optimization (considering each piece of cubic separately) and then in the global optimization. We have experimented also with another minimization routine called **fminu.m**. The first is based on the Nelder-Mead simplex (direct search) method and the latter on the BFGS Quasi-Newton method with a mixed quadratic and cubic line search procedure. Here we compare the cpu and number of function evaluations in **gpsfit2** using **fmins.m** (called **gpsfit2s.m**) versus that using **fminu.m** (**gpsfit2u.m**):

The results for **gpsfit2s.m** are given in the following tables, first for the local optimization part (Table 13) and then for the global optimization (Table 14):

Data file name	Number of function evaluations	Function value
p2sep1.mat	108 45 39	.00044097 .00176556 .000438411
p10jul162.mat	34 61	.00091 .00131
a1p11c9.mat	30 58	.00049 .00026

Table 13: Local Optimization Using gpsfit2s.m

For the global optimization part **gpsfit2s.m** requires 1201 sec using data file p2sep1.mat.

Data file name:	p2sep1.mat	p10jul162	a1p11c9
Number of function evaluations:	1204	259	251
Function value:	.000693999	.00057	.0060
RMS	.71057	.4775	.4762

Table 14: Global Optimization Using gpsfit2s.m

Now to **gpsfit2u**. For the local optimization part we summarize the result in Table 15:

Data file name	Number of function evaluations	Function value
p2sep1.mat	14 26 10	.000441091 .00176124 .000448282
p10jul162.mat	15 26	.00091 .00131
a1p11c9.mat	9 26	.00049 .00026

Table 15: Local Optimization Using gpsfit2u.m

For the global optimization (see Table 16) **gpsfit2u.m** requires 170 sec for the global optimization part using data file p2sep1.mat (compare to 1201 sec for **gpsfit2s**)

Data file name:	p2sep1.mat	p10jul162.mat	a1p11c9.mat
Number of function evaluations:	169	129	64
Function value:	.00111341	.00055	.00067
RMS	.90676	.4621	.5011

Table 16: Global Optimization Using gpsfit2u.m