

Track Model Constraint Enhancement for NovAtel's OEM4

Tom Ford, NovAtel Inc., 1120-68th Ave., N.E., Calgary, Alberta, T2E 8S5, tford@novatel.ca
Ken Milnes, SportVision, kenmilnes@sportvision.com

BIOGRAPHIES

Tom Ford is a GPS specialist at NovAtel Inc.. He has a BMath degree from the University of Waterloo (1975) and a BSc in survey science from the University of Toronto (1981). He did research in the inertial and GPS field at Sheltech and Nortech surveys before becoming a member of the original group of GPS developers for NovAtel Communications (now NovAtel Inc.). He has helped develop many of the core tracking and positioning technologies at NovAtel Inc. Over the past few years his focus has been to develop and implement inertial/GPS algorithms for an integrated INS/GPS system. In addition he has been developing specific GPS only positioning algorithms such as the one described in this paper.

Ken Milnes is a senior scientist and project manager at Sportvision, Inc. As a co-founder of Etak Inc., he performed extensive work in dead reckoning and digital map matching algorithms for land navigation. At SRI International, he did research on HF radar systems. He received a BS in Electrical Engineering and Computer Science from University of California at Berkeley.

ABSTRACT

The use of clock and position constraints is a standard part of GPS navigation. A 3 dimensional position constraint is normally used to facilitate the generation of differential corrections, and to provide a fixed location to which RTK vectors can be applied so a precise position may be computed. Furthermore, clock and height constraints can be used to improve the geometry provided by the satellite constellation and in some cases provide a degraded solution in cases when less than 4 satellites are available. But in land and air applications, such constraints are not particularly useful for navigation because the constraints are often not accurate enough to strengthen the navigation solution significantly.

Recently, NovAtel Inc. was approached by SportVision with a request for a reasonably priced navigation system which could provide 1 metre positioning accuracy on a racetrack where the satellite geometry would periodically be very poor due to restricted visibility. Initial attempts to use clock constraints, height constraints, a position velocity filter and a low cost inertial system did not meet the requirements. A novel approach using planar section constraints provides the required accuracy and in fact

facilitates a dramatic improvement in not only ambiguity resolution time but also ambiguity resolution reliability.

Briefly, the method requires a contiguous set of planar sections which represent the surface of the track on which the vehicle is driving. When a roving receiver is on the track, it searches for an appropriate planar section by projecting its approximate position onto a horizontal reference frame used by the model. Having found the appropriate planar section, the remote receiver constrains its position in the direction normal to the planar section. To be useful, the position of the defining planar sections corners must be known to high accuracy in the same reference frame as the base station co-ordinates. In this case the planar sections are produced photogrammetrically and have an accuracy of about 10 cm.

The use of such planar constraints has a dramatic impact on both pseudo range accuracy and RTK availability. This paper describes the methodology used to provide planar section constraints and will show test results which demonstrate the accuracy and reliability improvements achievable with this method.

INTRODUCTION

Many land applications which use GPS are hampered by the restrictions imposed by buildings and natural impediments to the transmitted GPS signal. Often the GPS geometry is too poor to provide the geometrical strength required to generate the position accuracy the application requires. A particular example of this is the SportVision application which requires GPS positioning accuracy of 1 metre 95% of the time, in conditions of reduced satellite visibility in a highly dynamic environment. The reason for this positioning requirement is that SportVision's client television network needed real time vehicle positions of the participating vehicles for on-screen annotation of the race cars during the broadcast of the race.

The actual environment is any one of a series of NASCAR race tracks. Although each track is somewhat different, they share four common features. First, the visibility to all satellites is severely reduced at some point on the track due to the existence of a grandstand on at least one side of the track. Second, the availability of satellites is reduced and the remaining signals are corrupted by the proximity of a 10 metre tall overhanging steel and wire fence on the outside

edge of the track designed to keep vehicle debris out of the stands. Third, the track is not level, and in fact the cross track slope is not constant, varying by as much as 35 degrees between the straight sections and the curves. Fourth, the vehicle's linear velocity is typically 100 m/sec. A final constraint on the system was cost. In a typical race there are 40 vehicles, and any technology advances made to one car have to be made to all of the others, so the prospective solutions had to incorporate GPS signals only or GPS with a very low cost inertial unit.

SportVision and NovAtel Inc. investigated various approaches to this problem. The first option involved a height constraint based on the previous position. Unfortunately, the dynamics involved made predictions that would satisfy the accuracy requirements impossible. A second approach included a clock constraint with an OCXO. Unfortunately, the clock model could not take into account the unpredictable effect of g sensitivity. The variability in the banking plus the centripetal acceleration experienced by the system made the lateral acceleration vector vary by up to 10 m/sec. So the clock predictions did not help the system satisfy the system's position accuracy requirements. We at NovAtel Inc. attempted to develop a decentralized position/velocity filter that accepted position and velocity inputs from the existing least squares filter and used these to generate an integrated model which could be used when the number of satellites in the constellation dropped below 4. Unfortunately, this didn't work either. The random walk velocity model is not adaptable to the dynamics in the race environment, and random walk excursions of tens of metres were not uncommon. SportVision in conjunction with the manufacturer of a low cost inertial sensor, attempted to quickly develop an integrated solution, but unfortunately these kinds of system developments take more time than SportVision had.

In spite of all of these failures with the NovAtel Inc. receiver (by itself and in conjunction with other technologies), a preliminary test broadcast in November 2000 showed enough promise to SportVision and their client network that they were confident that an incrementally improved GPS only system could provide the system accuracy needed in the application. Two approaches SportVision and NovAtel Inc. thought could be promising were the use of track model constraints and the use of on site pseudolites. Of the two, the track model enhancement was closer to production, so we decided to concentrate on developing it first.

The constraint methodology associated with a track model constraint is similar to the use of a height constraint. If a height constraint is used to aid the position estimation, the method used is to assume the constraint is with respect to a planar surface which is parallel to the local level plane at the approximate position of the receiver. Then the

uncertainty of the constraining position can be represented in the local level frame by a diagonal covariance matrix with large entries for the horizontal components and a relatively small entry for the vertical component. Since the estimation is done in the Earth Centred Earth Fixed (ECEF) frame, the covariance in the local level frame has to be transformed to the ECEF frame with the linear transformation relating the two frames. If the position changes significantly, the local level planar surface will change with respect to the ECEF frame and the transformation has to be repeated. Otherwise the same ECEF covariance can be used over again. In a track model constraining process the implementation is very similar except the constraint position and covariance (or weight) matrix changes at almost every positioning epoch, and the constraining planes are not necessarily parallel to the local level plane. The constraining method is applied to both the least squares and RTK processes, and the specific implementations for these processes are described later on in more detail.

Before proceeding with those descriptions, let's look at the track model representation itself. A track model is a set of planar surfaces which approximate the contiguous surface on which the navigation takes place. Each planar surface is defined by three vertex points (a triangle). To obtain this, a digital representation of the track surface is created using aerial photogrammetry techniques. High resolution overlapping aerial photographs are taken from approximately 300 meters above the track surface. Analytical photogrammetric workstations are used to create a digital terrain model of the track surface. Observable features such as paint, fences, walls, buildings, and pavement boundaries are captured as well. These features are used to verify proper registration of the track model with the differential GPS system. The relative accuracy of the track model is within 10cm. The constraint provided by this is that while the antenna is "within" the triangle, the position of the antenna is constant in the direction normal to the planar section. Based on a fixed antenna height, a planar constraint can be defined with respect to the local planar section. This is quite similar to a height constraint. Complexity in the implementation of the track model arises partly from the difficulty in determining which planar surface to use as a constraining surface, and partly from the system design required to generate the constraint for a real time system. This is made more difficult due to the fact that the internal GPS filter works in the ECEF frame, while the track model is described in the geographic frame. In order for the application to be successful in real time, a transformation to an intermediate planar reference frame is implemented. A description of this transformation, and the subsequent search process is included later.

The position/clock filter in the NovAtel Inc. series of GPS receivers is based on a single epoch least squares technique.

The first approach was to modify the least squares routines so they could accept a planar section constraint and associated weight matrix in its estimation process. This required very small changes to the estimation process, was successfully implemented in a short time and showed a marked improvement in the system's positioning accuracy. These results are shown later on.

After the pseudo range filter modifications and testing, it occurred to me that the constraint process would be ideally suited to aid the Real Time 20cm (RT20) filter. The benefit to this filter would be that the weakest direction (up) would now have an initializer with significant weight. It can help the RT20 filter provide more accurate positions as well as providing an accurate seed sooner to the RT2 filter. So the methodology used to constraint the RT20 filter is described and real time test results for that filter are included.

TRACK MODEL DETAILED DEFINITION

The track model positions are defined in WGS84 geographic co-ordinates but the internal reference frame for the GPS filter is in ECEF co-ordinates. This would not be a problem (the geographic co-ordinates can be simply transformed to ECEF (xyz) vectors, except that the triangle search engine requires a primarily two dimensional frame. This could be satisfied if the internal position was transformed to geographic co-ordinates, but this transformation is time consuming, and it is possible it may have to be carried out more than once per solution. So a different approach was to generate a local plane representing the model and a simple transformation that converts vectors in the ECEF frame to vectors in the local model frame. The corner positions of all the triangles (in the ECEF frame) are differenced with a local "base position". These are rotated to the local frame by the rotation matrix required to rotate a vector in the ECEF frame at the base position to a vector at the base position but in the geographic frame. So local co-ordinates are generated in this manner for all the points in the track model. The generation is simple

Co-ordinates of model point in the local frame:

$$P_l = R_e^{-1} * (P_{ECEF} - P_{BaseECEF})$$

Where $P_{BaseECEF}$ is the base position vector in the ECEF frame,

P_{ECEF} is the track model position in the ECEF frame,
 R_e^{-1} is the rotation matrix used to transform a vector in the ECEF frame to the geographic frame at the base position.

If a triangle search is required, the current GPS position is transformed to the local frame via the same method and the search progresses as usual in that frame. It is possible that

additional optimisation regarding the search may be required once the algorithm is working on the OEM4 card.

Internally, the co-ordinates for all the points are maintained both in the ECEF frame and in the local frame. The constraint position is generated from the ECEF co-ordinates, and the search algorithm is applied using the co-ordinates in the local frame. The search algorithm described later finds an appropriate triangle. The previously generated constraint position is taken from it and used as a seed position in the filter. The 6 weight matrix elements for that triangle constraint are expanded to generate a P_x matrix for the least squares solution.

SEARCH MECHANIZATION

The track model consists of a series of triangles parameterized in a local planar reference frame. The search algorithm uses a routine supplied by SportVision which determines whether or not a planar (x,y) position falls inside or outside of a triangle. If it is inside, then that triangle is used for the constraint, otherwise it continues the search. This process continues until a triangle is found which satisfies the conditions, or all of the possible triangles have been eliminated as constraint candidates. The process is optimised by dividing the triangle into 256 rectangular regions parameterized in the same local planar co-ordinate system. It is easy to compute which of these rectangles includes a particular position. Then a restricted search of only the triangles within the target rectangle can be made. Therefore, a search which would normally require an average of 1000 triangle checks can be reduced to one which takes an average of 16 or so.

LEAST SQUARES IMPLEMENTATION

In the following, the least squares process and the height constraint modifications to it are described. The modification required to constrain to an arbitrary planar surface follows.

The least squares filter generates corrections to the system's ECEF position and clock according to the equation

$$\delta X = (A^T P A)^{-1} A^T P \omega$$

where

δX = correction vector to position vector and clock
 $[X, Y, Z, Clk]^T$

A = design matrix (nx4) based on satellite to receiver geometry

$$\text{In detail } A = [A_1, A_2, A_3, \dots, A_n]^T$$

$$\text{And } A_i = [\partial R^i / \partial X, \partial R^i / \partial Y, \partial R^i / \partial Z, 1]$$

$$\text{With } R^i = ((X^i - X)^2 + (Y^i - Y)^2 + (Z^i - Z)^2)^{1/2}$$

X, Y, Z = ECEF user position

X^i, Y^i, Z^i = ECEF satellite position

P = Pseudo range observation weight matrix (nxn) which is diagonal with the diagonal entries being the reciprocal of the variance entries of the pseudo ranges.

ω = The vector of misclosures between the theoretical observations based on the current satellite set and the last set of positions estimated, and the actual observations (pseudo ranges).

So

$$\omega = R_{obs} - R^i - Clk$$

$$= R_{obs} - ((X^i - X)^2 + (Y^i - Y)^2 + (Z^i - Z)^2)^{1/2} - Clk$$

At every observation time, the process is repeated until the length of the vector of corrections (δX) to the position/clock parameter vector is small enough. This is normally accomplished after two iterations. At each epoch, the previous position and clock estimate is used to start the process, but any covariance information associated with that estimate is ignored. This means that at every epoch, at least 4 satellites are needed to estimate the 4 elements on the position/clock vector. If information related to the position/clock parameters were available, then this could be included in a modified least squares process according to the following:

$$\delta X = (A^T P A + P_x)^{-1} A^T P \omega$$

where

P_x = Parameter weight matrix (4x4) based on knowledge of the parameters includes in the estimation process.

If certain elements of the parameter vector are well known, then this knowledge can be incorporated in the system by making the appropriate diagonal elements of the parameter weight P_x large. If, for example, the clock estimate has a standard deviation of 1/2 m, then the P_x entry $P_{4,4}$ would be 4, and one less satellite would be required in the estimation process to generate a 4 parameter solution.

There are more complications if the knowledge of height is to be represented by this system because height is in the geographic reference frame and so the covariance information for height must be transformed from the geographic frame to the ECEF frame before it can be used by the system in the estimation process. But this is fairly easy because the transformation between the two frames is well defined.

For this case, the P_x matrix is

$$P_x = C_x^{-1} = (J^T C_g J)^{-1}$$

Where

C_g = The covariance matrix of the position/clock in the geographic frame.

J = The matrix of derivatives of the transformation of position/clock from the geographic to the ECEF frame.

C_x = The covariance matrix of position/clock in the ECEF frame.

In the case of the track model application, J is not the rotation matrix used to transform a vector from the geographic to the ECEF frame, but instead a rotation matrix used to transform a vector from the planar section frame to the ECEF frame. This matrix is easy to generate by simply representing three basis vectors, describing the planar section frame and a normal to it, in the ECEF frame. The positions of the vertices of each triangle are transformed from the geographic to the ECEF frame. The differences of these vectors are parallel to the planar section, and the cross product of two of these difference vectors provides a normal vector to the planar section. The cross product of the normal vector with either of the vector differences generates a vector parallel to the planar section and orthogonal to the other two vectors used in the cross product. Finally, normalising these three vectors provides a set of orthonormal basis vectors representing the planar section frame in ECEF co-ordinates. So this set of vectors can be combined to generate J , the 3 by 3 rotation matrix used to rotate a vector from the planar section frame to the ECEF frame. Symbolically:

$$J = [B_1 | B_2 | B_3]$$

Where B_1, B_2, B_3 are the basis vectors whose construction is defined in the previous paragraph.

The constraint position is given by the average of the three corner positions in the ECEF frame plus the constraint position relative to the planar section, transformed to the ECEF frame. Symbolically, this is:

$$\text{Constraint position: } P_{cp} = ((P_1 + P_2 + P_3)/3.0) + J * [0,0,h_a]$$

Where P_1, P_2, P_3 are the ECEF positions of the planar section corners, and h_a is the antenna height with respect to a level planar section.

This is fairly straight forward, and mimics the constraint logic used to generate geographic height constraints.

RT20/RTK FILTER MODIFICATION

RT20 [1] is a Kalman filter that generates estimates of the relative position between a reference and rover receiver as well as estimates of floating ambiguities related to the double difference carrier observations for those two receivers. In the NovAtel Inc. receiver, RT20 provides a best available solution when RTK is not available as well as providing an initial search space for the RTK carrier based process [2].

Carrier positioning is a process in which a relative position between two ground sites (a base station and a remote station) is computed based upon observed fractional phase differences and known whole cycle differences between the two sites. The fractional and whole cycle differences

together produce a synthetic observation which is equal (when converted to metres) to the geometrical difference in distance between the two sites and the satellite they are both observing. Knowledge of the whole cycle portion of the synthetic observation cannot be determined directly from the observations, but must be determined indirectly from many observations over time during what is known as a whole cycle resolution process. The whole cycle difference is also known as a carrier ambiguity, and the resolution process is known as an ambiguity resolution process.

In order to resolve fixed integer ambiguities, an initial guess of the position difference is made and a series of sets of ambiguity candidates is selected such that each set will generate a position difference that is close to the one chosen in the initial guess. Each set is used to compute a position difference and an associated set of residuals. For each set, these residuals are accumulated and the accumulation compared to a theoretical accumulation and also to other accumulations in the series of candidate sets. If the correct set of ambiguities is in the series, then eventually its residual accumulation will be close to the theoretical accumulation and also smaller than any of the residual accumulations for the other sets. At this time the correct ambiguity set is known and can be used to generate relative positions with carrier type accuracy.

To summarise, there are two things that have to be done to resolve ambiguities:

- 1: Guess at an initial position, and an associated search space whose size is based on the precision of the initial position estimate.
- 2: Use the guess and its precision to define a series of candidate sets of ambiguities and then accumulate computed residuals over time and eliminating sets whose residual accumulation exceeds some kind of threshold.

Typically a Kalman filter with both position and ambiguity states is used to define an initial guess for the search space. It is run in real time as carrier and pseudo range observations are provided to it and some kind of executive routine monitors its position covariance to see when the search space can be defined and search can commence. By including position constraints with the GPS observation set, the precision of the initial position estimate used to define the search space can be reduced sooner and more, and this should significantly speed up the resolution process.

The Kalman filter used to estimate position and floating ambiguity states can be described as follows:

State: $X=[x,y,z,N_1,N_2,\dots,N_k]$

State Initial Covariance: $P=$ [big diagonal elements, 0 off diagonal elements]

The design matrix H defines the linear relationship between the double difference observation (satellites r,j and the two receivers) and the state elements.

For satellite j and reference satellite r the phase relationship is

$$H = [\Delta x_m^r/R_m^r - \Delta x_m^j/R_m^j, \Delta y_m^r/R_m^r - \Delta y_m^j/R_m^j, \Delta z_m^r/R_m^r - \Delta z_m^j/R_m^j, 0, 0, \dots, 1, 0, \dots, 0]$$

While the pseudo range relationship is

$$H = [\Delta x_m^r/R_m^r - \Delta x_m^j/R_m^j, \Delta y_m^r/R_m^r - \Delta y_m^j/R_m^j, \Delta z_m^r/R_m^r - \Delta z_m^j/R_m^j, 0, 0, \dots, 0, \dots, 0]$$

The Kalman filter mechanization is as follows:

$$\text{Gain: } K_k = P_k (-) H_k^T [H_k P_k (-) H_k^T + R_k]^{-1} I$$

$$\text{Covariance Update: } P_k (+) = [I - K_k H_k] P_k (-)$$

$$\text{State Update: } X_k (+) = X_k (-) + K_k [Z_k - H_k X_k]$$

Where

R = Observation covariance matrix (scalar for phase and pseudo range observations)

z = Observation (pseudo range or carrier measurement)

In the pseudo range and phase measurement implementation, the observations are decorrelated and the updates are done serially, one for each observation. With the position constraint information from the track model, the observation/state relationship is very simple

$$H = \begin{bmatrix} 1, 0, 0, 0, \dots, 0 \\ 0, 1, 0, 0, \dots, 0 \\ 0, 0, 1, 0, \dots, 0 \end{bmatrix}$$

$H = [I, 0]$ with $I = 3 \times 3$ and $0 = 3 \times (n-3)$, (n = number of states)

And C_x is the covariance matrix of the constraint position:

$$C_x = J^T C_t J$$

Where

C_t = The covariance matrix of the position in the "triangle" frame.

J = The matrix of derivatives of the transformation of position from the "triangle" frame to the ECEF frame.

$$C_t = \begin{bmatrix} 10000, & 0, & 0 \\ 0, & 10000, & 0 \\ 0, & 0, & 0.01 \end{bmatrix}$$

that is, the parallel elements are more or less unknown, and the normal element is known to 10 cm at 1 sigma.

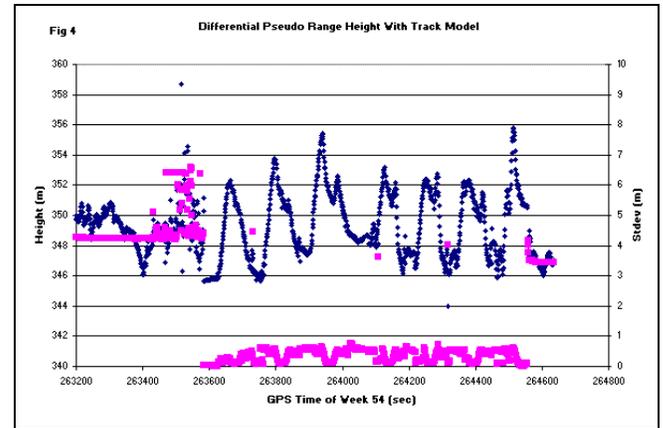
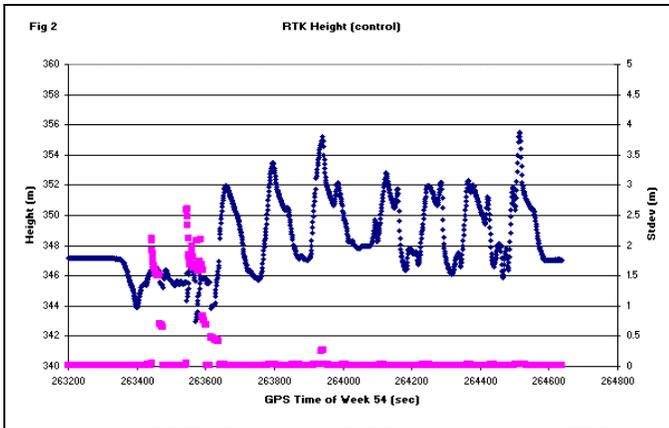
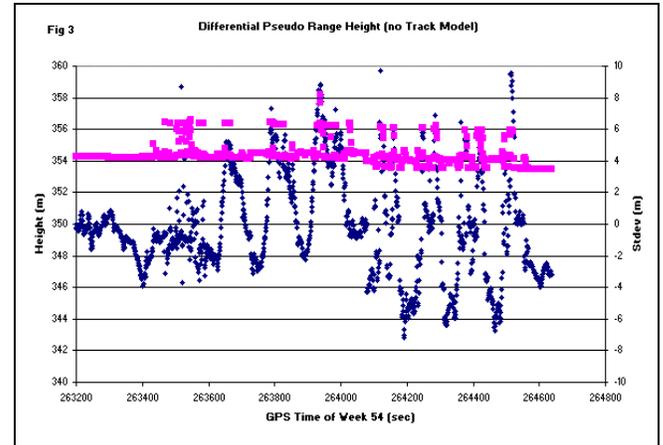
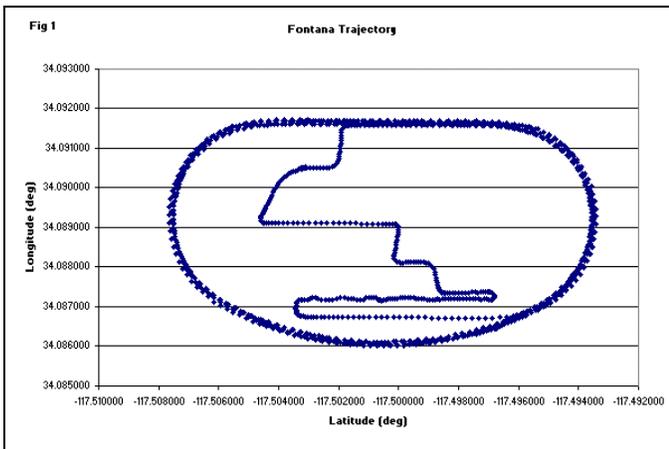
This completes the description of the RT20 process and the track model constraint modification to it.

TEST RESULTS

The results shown here are based on data collected during two tests at the Fontana race track (California speedway) in Ontario California. The first set of results compares the performance of the system in single point least squares mode with a track model and without. The data for this was collected in September 2000, and the results were generated with a post mission process. After the RT20 filter was modified to use track model constraints, the same data was processed with the modified RT20/RTK process. The second set of results are based on a real time data set (that is generated in real time with a track model inserted into the OEM4 software), and address the effectiveness of the system operating in RT20/RTK mode.

LEAST SQUARES FILTER TEST RESULTS

The test data used in the following analysis was collected at the Fontana track in California on September 5, 2000. The tracking environment was excellent, so that continuous RTK positions are available. These are used as control positions against which the unaided and aided differential pseudo range positions can be compared. Since height is the most sensitive component to the track model constraint process, the focus is on the height results. There are four plots to follow. The first shows the trajectory of the test vehicle around the Fontana track. The second shows the RTK height and standard deviation over time. The third shows the same data generated from unconstrained single point pseudo range data, and the last shows the heights from a single point pseudo range process when the process is aided with track model constraints.



The aided results shown on Fig 4 indicate a significant improvement in height accuracy compared to those shown in Fig 3. The unaided height shows a typical standard deviation of between 3 and 4 metres, while the aided heights have standard deviations of typically 0.5 metres or

less. The heights shown on Fig 4 before time 263600 and after 264550 correspond to infield positions for which the track model data did not exist.

RT20/RTK FILTER TEST RESULTS

The success or failure of the use of track model constraints depends on the reliability of the model, and as it turns out, the consistency of the datum used for the model and the differential reference station. So a preliminary section which addresses the reconciliation of the two datums is included prior to the RT20/RTK results.

The model to base station datum discrepancy is determined below by comparing the track model normal constraint with the precise GPS position in the direction normal to the track model section applicable to the GPS position. This comparison is given by ω , the misclosure used later in the RT20 filter.

$$\omega = R_c^P (Row\ 3) (POS_{RT} - POS_{TM})$$

Where

R_c^P = the rotation matrix used to transform a vector from the ECEF to “triangle” frame

POS_{RT} = the GPS ECEF position (either RT20 or RT2)

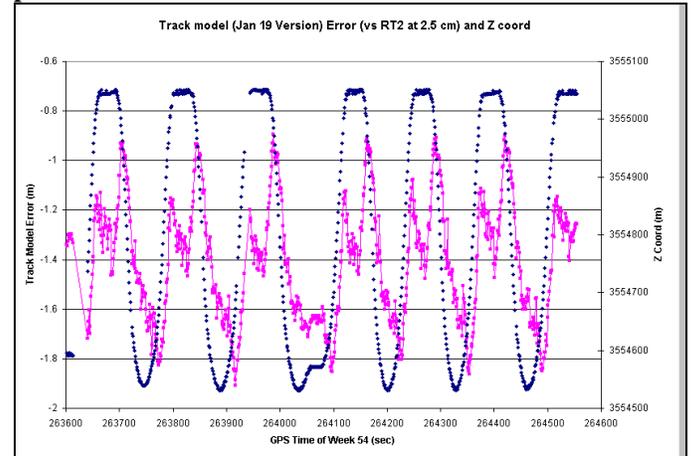
POS_{TM} = the track model constraint position in the ECEF frame

Note that ω is just the third element of the vector, because this is the part in the direction normal to that pertinent triangle.

The following sets of plots documents the validation work on the track model supplied by SportVision. The misclosure indicates the accuracy of the constraint. Comparisons between an updated model and Real Time 2 cm (RT2) are shown in Figures 5 and 6. There are significant errors dependent on the position of the vehicle on the track shown on Figure 5. These errors are primarily related to the use of base station coordinates referenced to a different datum than the track model coordinates. Figure 6 shows results when both the track model errors and base station inconsistencies are removed. The inconsistencies (datum shift) between the base station and track model datums are observable via the misclosures described above, and a method is described below that can be used to estimate these shifts.

The following (Fig 5) shows a normal direction comparison of the track model with RT2 vectors. Only RT2 positions with standard deviations of less than 2 cm are used in the comparison.

Fig 5 Track Model/RTK normal direction misclosures prior to reconciliation



Based on the available data, the following estimation process can be used to determine the offsets required to reconcile the base station and track model reference frames:

The offset between the base station frame and the track model frame is reflected in triangle frame coordinates as $x_3^t = x^e \cdot n_3$

The observation equation that models this vector component is

$$\omega = x^e \cdot n_3 = R_c^P (Row\ 3) (POS_{RT} - POS_{TM})$$

or

$$\omega = x^e \cdot n_3 = U_3^T R_p^e (POS_{RT} - POS_{TM})$$

Where

x^e = Base station shift in the ECEF frame,

x_3^t = z component of base station shift in triangle frame

n_3 = normal vector to the triangle in the ECEF frame,

R_p^e = the rotation matrix used to transform a vector in “triangle frame” coordinates to the ECEF frame

U_3 = unit vector normal to the triangle in the “triangle frame” $U_3 = [0,0,1]^T$

and \cdot = dot product operator

Noting that n_3 is simply the transpose of the last column of R_p^e , a least squares estimate can easily be generated from this ω via

$$X = (\Sigma(A^T A))^{-1} \Sigma(A^T \omega)$$

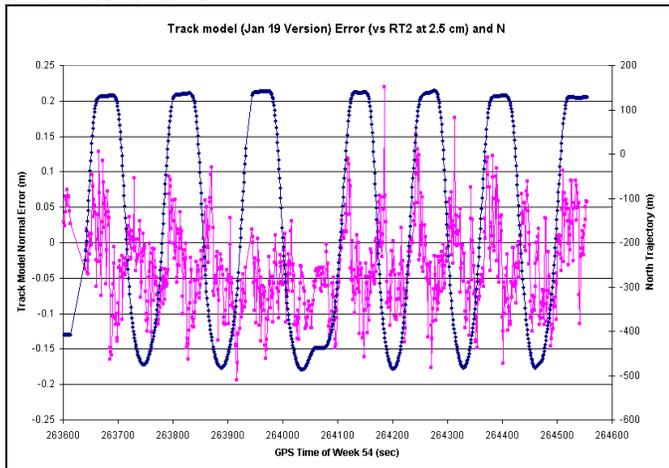
Where

$$A_i = n_{3i} = R_p^e U_3$$

The summation goes from $i = 1$ to the number of RTK observations on the model. In order for this to work, a model with reasonable variation of normal vectors has to be used if all three components are to be observable.

The method was implemented and the estimated base station coordinate shifts were applied to the base station coordinates to generate the improved results shown on Figure 6 below.

Fig 6 Track Model/RTK normal direction misclosures after reconciliation



There are still some position related systematic errors evident on this plot, but the residual errors could result from errors in the photogrammetry. A triangle specific height shift (observable with RT2 positions) could remove this error, but as it turns out, the improved performance based on a 10 cm track model error is sufficient.

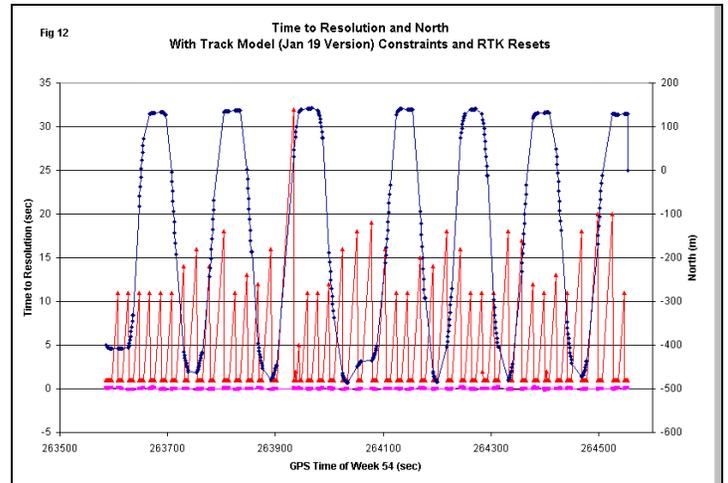
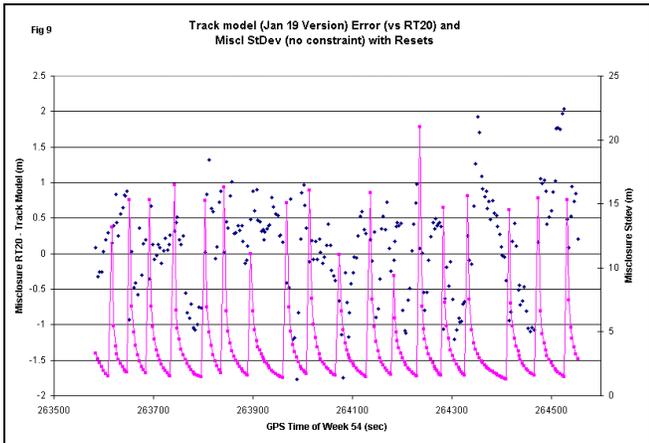
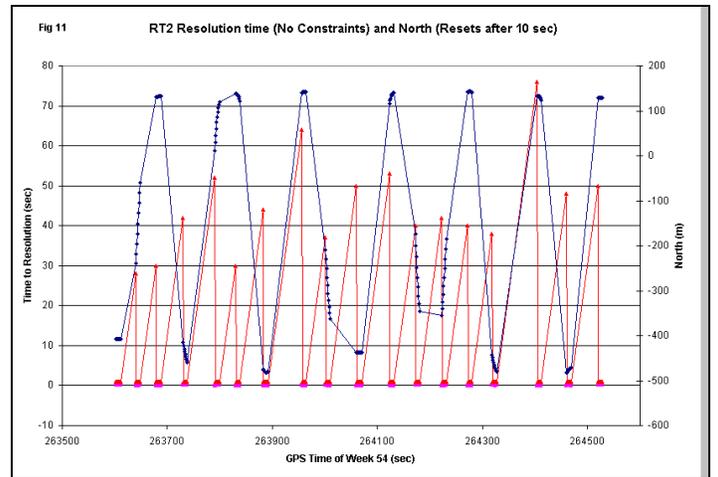
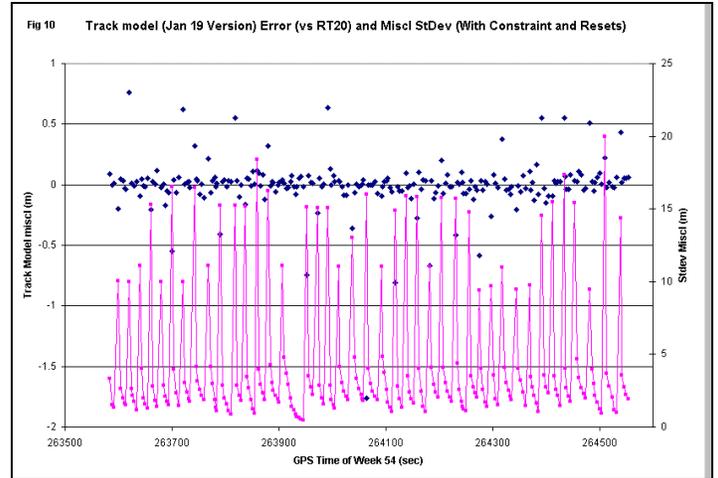
RT20 PRELIMINARY TEST RESULTS:

The Fontana data collected last September are used to generate constrained and unconstrained position results. The items of particular interest were resolution reliability and the time to resolution possible when constraints are available compared to when they are not. First lets look at the improvement in resolution time. In order to see this, I forced the ambiguity filters (both RT20 and RT2) to reset 10 seconds after every resolution, and then allowed the ambiguities to be re-resolved.

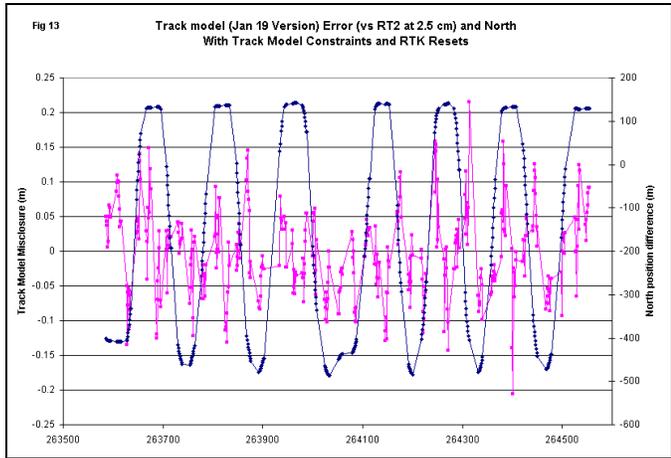
The first plot (Fig 9) shows the standard (unaided) RT20 misclosure from the track model and the misclosure standard deviation against time. After each resolution the filter is reset and the misclosure standard deviation starts at the 10 to 15 metre level and decreases over the next 40 to 50 seconds to the 1.5 metre level.

The second (Fig 10) shows the same data but with track model aiding the RT20 process. The standard deviations shown are before the update, and that is why they are so large. The unconstrained system managed 19 resolutions (average resolution time of 51 seconds), while the aided system completed 4 resolutions, for an average resolution time of 24 seconds.

Figures 11 and 12 show the time between successive RT2 records as well as the north co-ordinates of the RT2 positions only. The long time intervals show the resolution times.



The Figure 13 shows the track model misclosures of the resolved positions. The misclosures are identical (when available) to the ones seen in Fig 6, and this indicates that the 41 resolutions are correct.



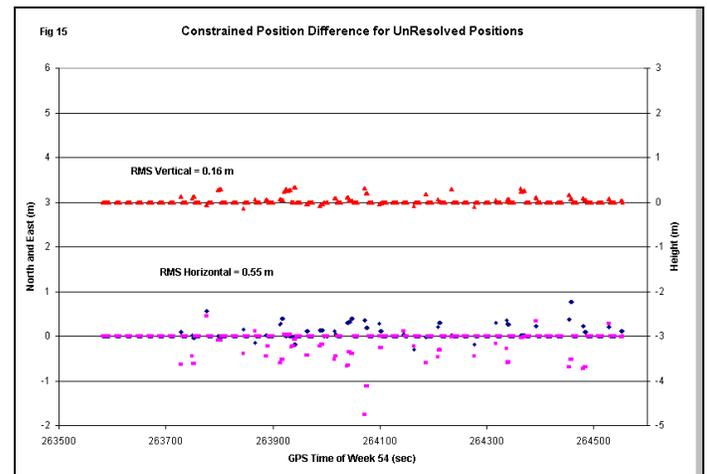
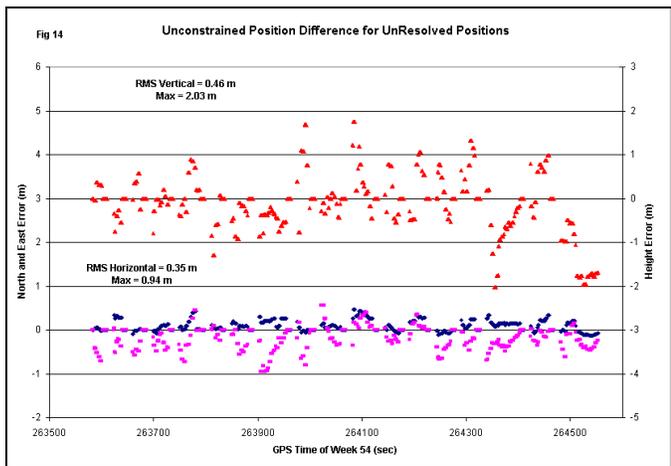
The errors in position that occur in the resolved constrained position are the same as the errors in the resolved unconstrained position, that is typically about 2cm. This

verifies that the resolution process is successful in the constrained case.

If the ambiguities are not resolved, then the position error will be larger. The difference between RT20 positions with RT2 positions effectively generates an error set for the RT20 positions. A comparison of the differences between these errors in the unconstrained case (Fig 14) compared to the constrained case (Fig 15) highlights the advantages of the track model constraints in the unresolved case. The following Table 1 summarizes the results shown in Fig 14 and 15. Since the track model is primarily a height constraint, the vertical errors in the constrained case are much smaller. The large horizontal error in the constrained case is a result of poor geometry, and the standard deviations of the position at this time reflects this. If the unconstrained system had dealt with the same geometry in the same mode, then its error would have been high there too. The reason the RT20 RMS is higher for the constrained case is because so much more of the positions computed are in the high variance portion of the floating ambiguity convergence curve. So the RMS appears worse because the constrained system spends less time in RT20 mode. Note that the unconstrained case has more samples in RT20 mode because it took much longer to resolve ambiguities.

TABLE 1: Post Mission RT20 Only Performance Statistics

Case	Samples out of 972	Horizontal RMS	Horizontal Max	Vertical RMS	Vertical Max
Unconstrained	628	0.35 m	0.94 m	0.46 m	2.03 m
Constrained	146	0.55 m	1.75 m	0.16 m	0.33 m



FONTANA FIELD TRIALS FEB 13,14,15, 2001

On Feb 12, 2001, Mike Bobye and I met Michael Brown in Ontario, California, the site of the California Speedway (Fontana racetrack) for a race track trial. The object of the trial was to validate the modifications made to the OEM4 software in a real time test. For successful validation, 4 criteria had to be met:

1. The receiver must have no serious system level degradations, including memory errors or significant CPU overloads.
2. The accuracy of the constrained system had to be better than that of the unconstrained system.
3. The resolution time of the constrained system had to be less than that of the unconstrained system.
4. The reliability of the constrained system had to be better than that of the unconstrained system. In other words, there had to be fewer incorrect resolutions.

In order to do this, a comparison of the results from a modified OEM4 was made with the results from a standard OEM4 configured as an RTK receiver. The two receivers processed signals received from a single antenna mounted on the roof. The base station telemetry line was split so both receivers had access to identical RTCA range and phase observations.

The Fontana track is an oval track (see Fig 1) approximately 3.5 km long, with banks of about 20 degrees on the east and west ends. A 40 metre high grandstand extends the length of the south side of the track and provides decent obstruction to the receivers of satellites in the southern sky when the receivers are on the south side of the track. Between the track and the grandstand is a wire mesh fence designed to catch debris that results from normal and abnormal race conditions. The fence is 7 metres high and extends about 3 metres over the edge of the track. It consists of 15 cm square wire mesh and generates significant perturbations with the GPS signals. The following picture shows the fence and a portion of the track.



The track itself is 16.8 metres wide and is divided into 5 sections of 3.4 metres. For the purposes of test control we tried as much as possible to follow the divisions between the pavement sections. Then height comparisons from one lap to the next can be made and good agreement should indicate the system is working better than if the lap to lap comparison is not good. There were 4 divisions or rows, which are labeled Row 1 to Row 4. Row 1 is next to the infield, and is relatively unobstructed. Row 4 is adjacent to the fence (3.4 metres away), and with the fence overhang, exactly $\frac{1}{2}$ of the sky is at least partially obstructed. Row 1 can be seen under the car in the following picture.



An important part of this test was the generation of base station co-ordinates that were in datum agreement with the track model itself. As described earlier, the base station shift can be solved by a least squares procedure that uses the difference between track model positions and positions generated as the sum of RTK vectors and the base station position. In this test the base station position was on top of the grandstand at an initially arbitrary point. An approximate WGS84 position was generated for it by averaging for $\frac{1}{2}$ hour or so. Using this position, an RTK data set representing several laps of the track (although 1 lap would have been quite sufficient) was generated with the standard OEM4 receiver. Using the previously described least squares procedure (two iterations were required), a base station shift of about 3 metres was computed for the base station. I should point out that during the initial set-up on Feb 13, Ontario was doused with the most single day rainfall of the last 10 years and our laptop computer with the base station shift software on it was damaged from the moisture. As a result we had to have the code emailed from Calgary and we had to buy a C++ developer kit from a vendor in Ontario, and so the base station shift could not be computed until the afternoon of Feb 14. So before the developer kit was available, we tried running the track model OEM4 with the approximate base station co-ordinates. The result of this was a system that generated height discrepancies of 3 metres (this was the

only observable component in the field) and a high frequency of RTK resets.

TEST DEFINITION

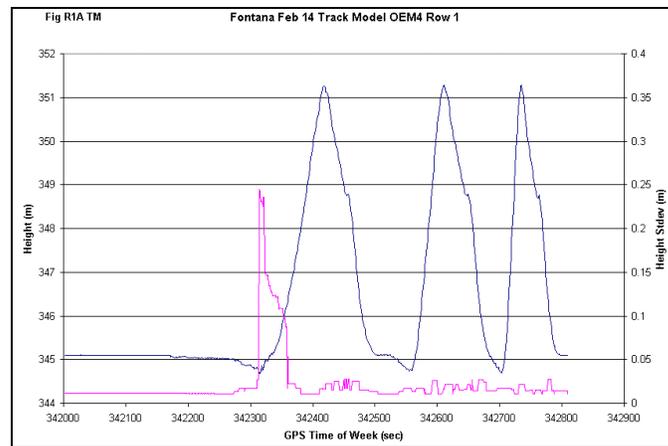
The test itself took place over 2 days, Feb 14 and 15. On the first day each of the “rows” were driven 3 times without any artificial resets.

On the second day each row was driven 11 times with the standard OEM4 and track model OEM4 combination, but on rows 1 and 2, filter reset commands were issued to the filter to re-initialize the carrier measurement ambiguities. The object of this was to measure the difference in ambiguity resolution times for the 2 receivers when the constellation was good. For rows 3 and 4 it was not necessary to issue reset commands because signal blockages in those rows caused the filter to reset anyway.

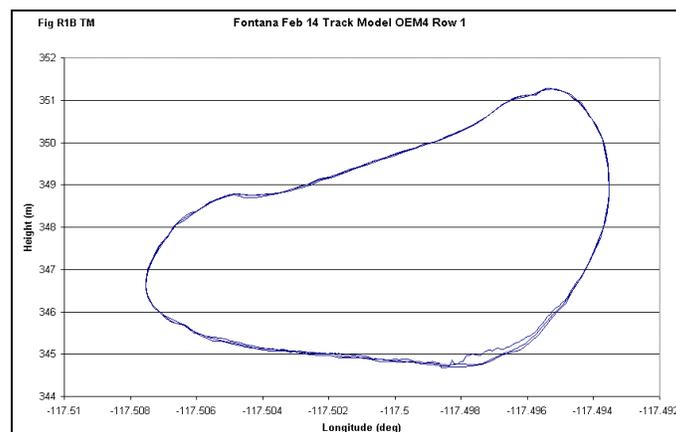
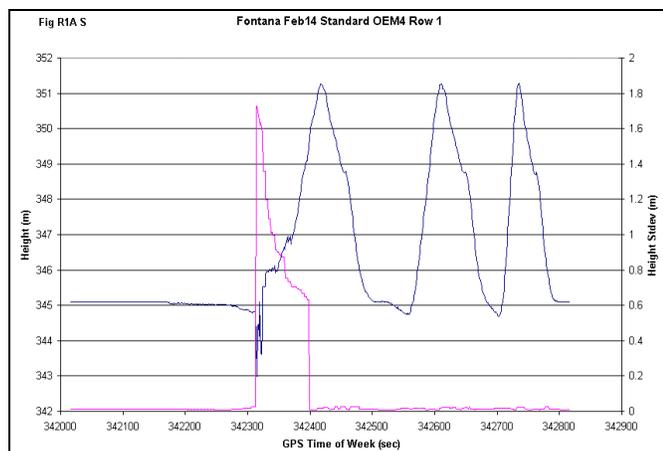
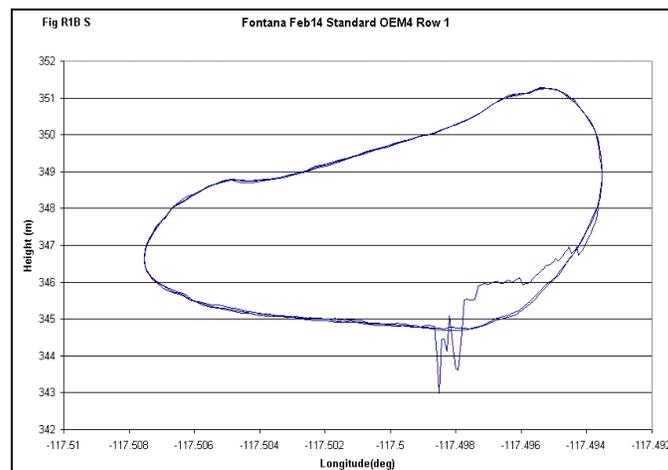
TEST RESULTS FEB 14

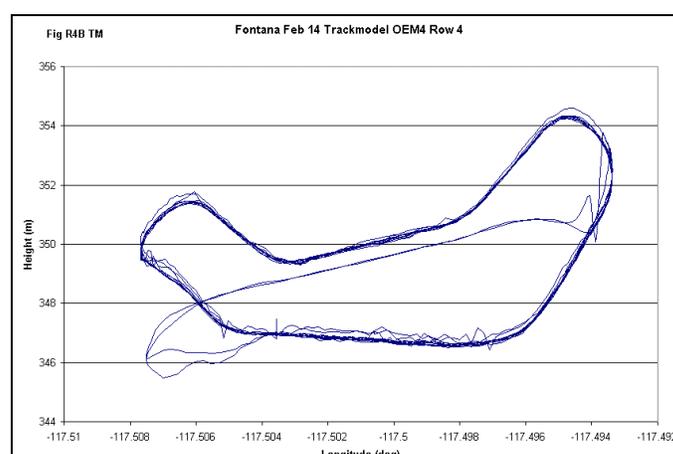
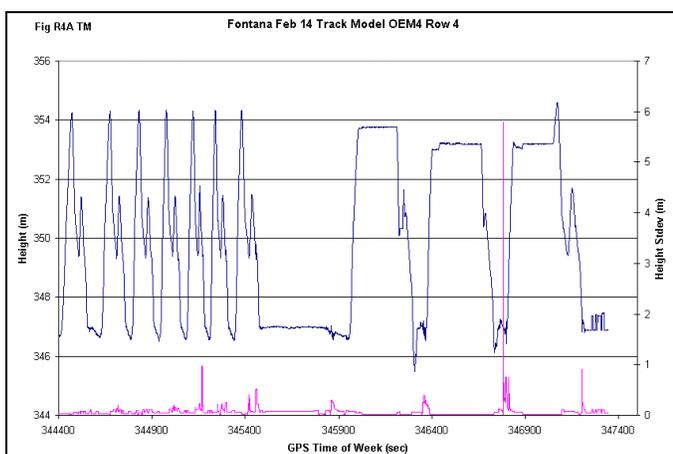
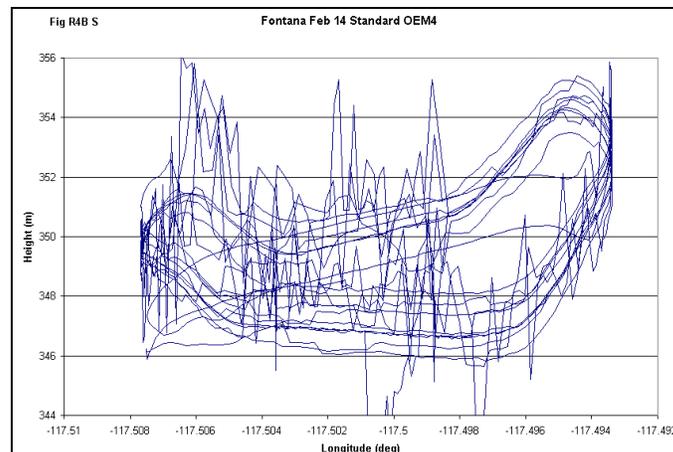
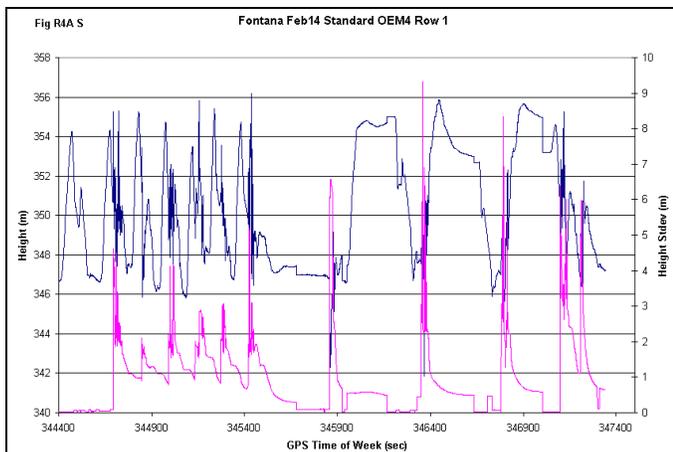
The Feb 14 results show fairly consistent results between the data collected on Rows 1, 2 and 3 but some discrepancy in Row 4.

The effect of an early RTK reset is seen on Fig R1A S and Fig R1A TM for the standard and track model system. The standard model height reaches a standard deviation of 1.7 metres, while the track model OEM4 height standard deviation is less than 0.25 metres. The time in RT20 mode for the standard system is about 90 seconds, while the track model OEM4 regains RT2 type positions after only 50 seconds.



The same Row 1 height data is shown on Fig R1B S and Fig R1B TM, but in this case plotted against longitude, so that the lap to lap repeatability can be seen more easily.





The Row 4 results above (“Fig R4B S” and “Fig R4B TM”) show a more dramatic difference in performance between the two systems, but this may be expected given the proximity of Row 4 and the fence. The first 7 laps were driven on Row 4 (3.3 metres from the fence), Fig R4A S and Fig R4B S indicate a significant degradation that occurs near the fence. The constrained system has resolved ambiguities almost continuously, but the standard system never reaches a fixed ambiguity state.

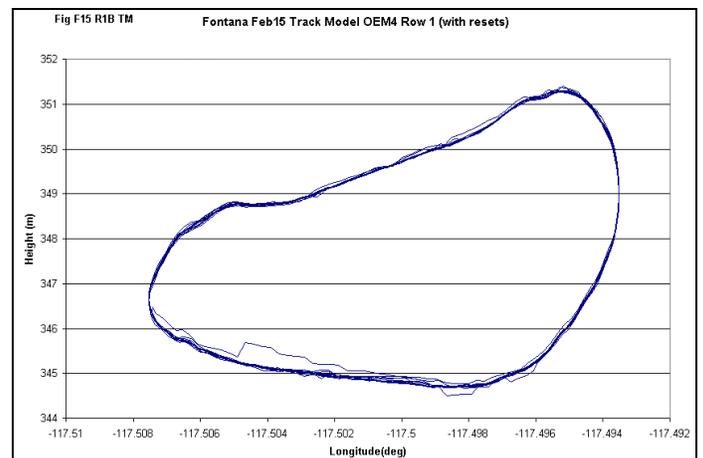
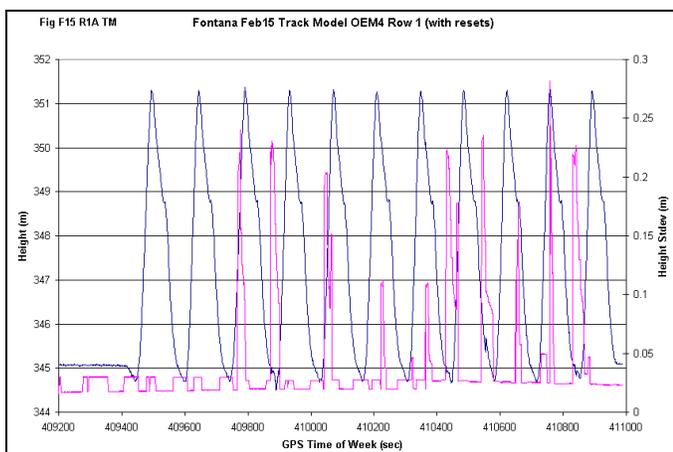
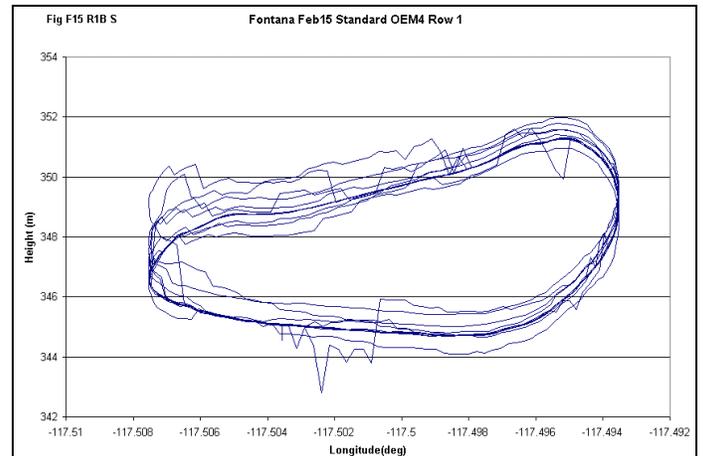
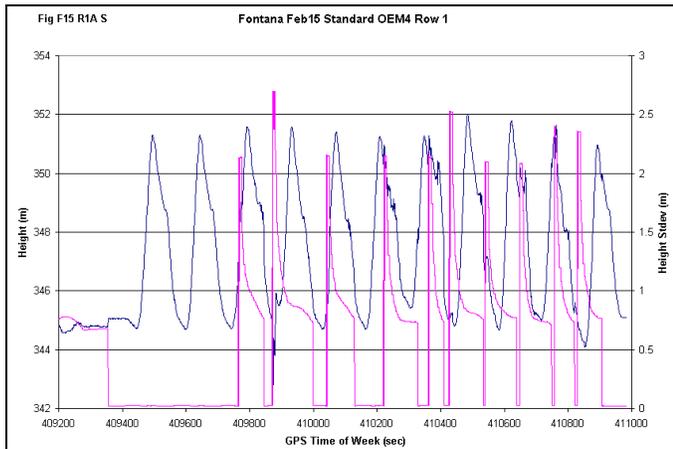
For the last three laps of the row 4 experiment we drove on the track between row 4 and the track wall, in what was the most obscured portion of the track. In this location we stopped and allowed the systems to resolve three times (not always at the same point on the track). During each of these resolutions, the standard OEM4 resolved improperly and the track model OEM4 resolved correctly as was verified by comparisons with control later on in the real time test. In addition, the number of satellites was about the same (between 4 and 6) for both receivers, and the idle time, which is a reflection of the CPU load, was between 50 and 60 percent. This indicates that there is no significant signal level degradation as a result of the track model constraint logic.

The plots above “Fig R4B S” and “Fig R4B TM” show the height vs longitude which gives a qualitative but indication of the improvement of the constrained version over the standard version. The excursions seen on the constrained plots occurred because we drove away from the row 4 resolution location to return via an unobstructed path during the resolution verification to a known control point.

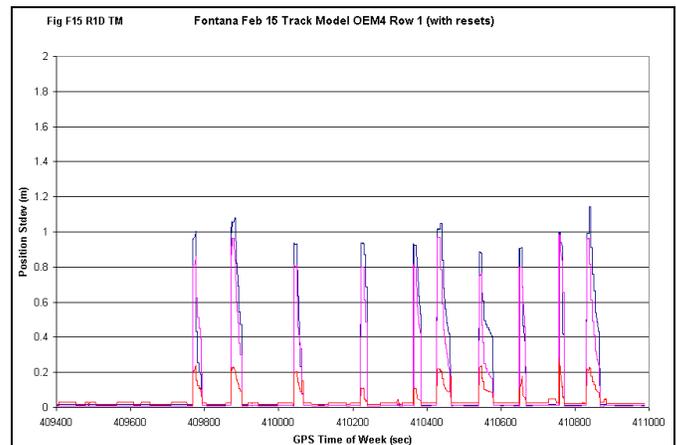
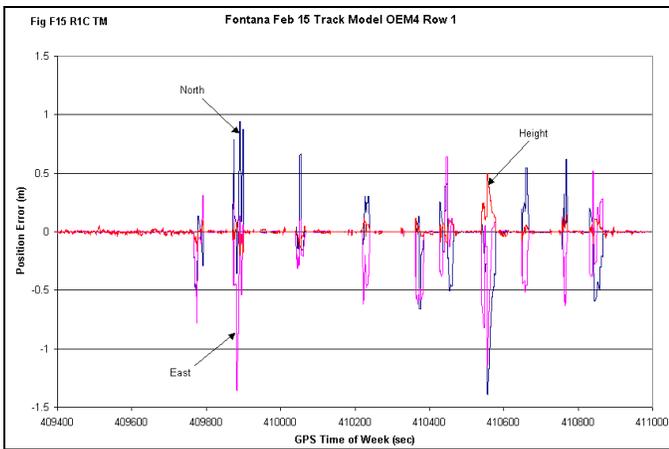
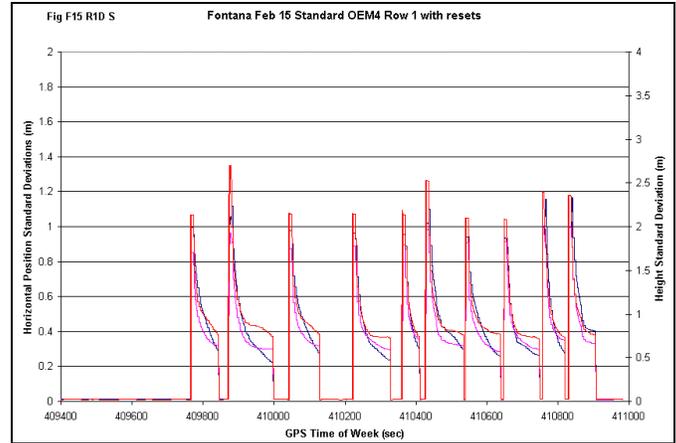
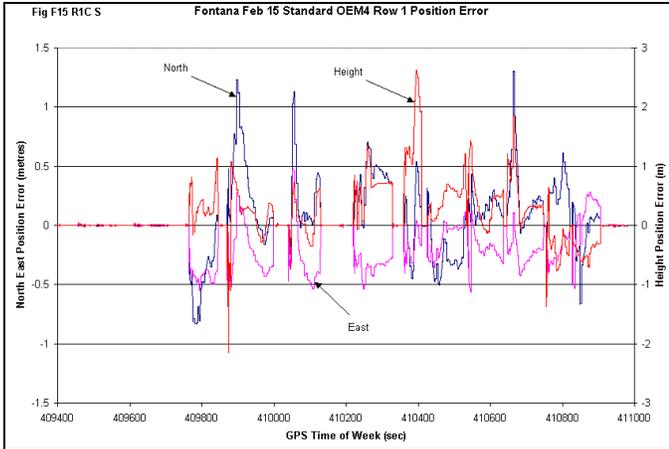
TEST RESULTS FEB 15

The testing on Feb. 15 was similar to the Feb 14 tests. We used each off the 4 pavement divisions at a driving guide in order to generate a repeatable path to follow on increasingly obstructed areas of the racetrack. As before row 1 is the least obstructed. The objectives of the experiment on this day are to determine the difference in resolution times between the two systems, to obtain accuracy estimates for the real time positions and to find out if the constrained system was resolving properly. For row 1 and 2 we issued reset commands after every resolution and then compared the resolved positions with a continuous trajectory constructed from the data reprocessed in post mission without any resets. Continuous tracking was not possible

for row 3 or 4, so repeatability is the only indication of the system's reliability.



These are row 1 results that show the effect of continuous RT2 resets on the two systems (standard system on top, height vs time on the left, height vs longitude on the right). In real time both receivers were allowed to resolve ambiguities, and then a reset command was issued. The resolution time in both receivers was not the same, so the commands were issued asynchronously. The commands were issued at different places in the track so the effect of different constellation shadowing could be observed. The effect of the reset command is to force the the receiver to discard all of the RTK ambiguity information (for both the float and fixed ambiguity filters) and return to pseudo range differential mode. The height repeatability is much better on the track model heights, and the resolution time is much less. The height standard deviation for the standard model in unresolved mode is between 2.5 and 0.7 metres, while the constrained height standard deviation varies between 0.25 and 0.1 metre in unresolved mode. The average resolution times for the standard and constrained system are 3 minutes and 25 seconds respectively. There were between 4 and 7 satellites tracked on both receivers.



The plots “Fig F15 R1C S” and “Fig F15 R1C TM” on the above left hand side show the difference between the real time positions with their reset induced errors and completely (and correctly) resolved positions generated without resets in post mission. As expected the heights on the track model OEM4 are more consistent than the heights for the standard OEM4, but the plots indicate that when the system is not resolved, the maximum horizontal error level is not too different for the two systems. However in the track model case the time the system stays in non-fixed ambiguity mode is much less, so the duration of the when the constrained system has large errors is much less. Also, the component with the highest error (by a factor of two), namely height, has been reduced significantly.

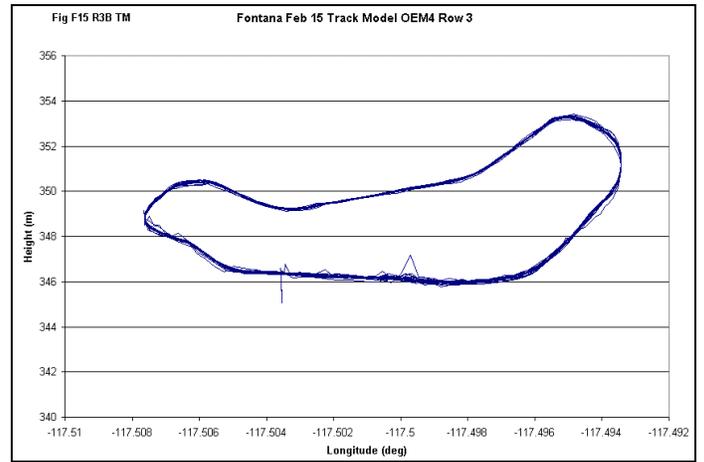
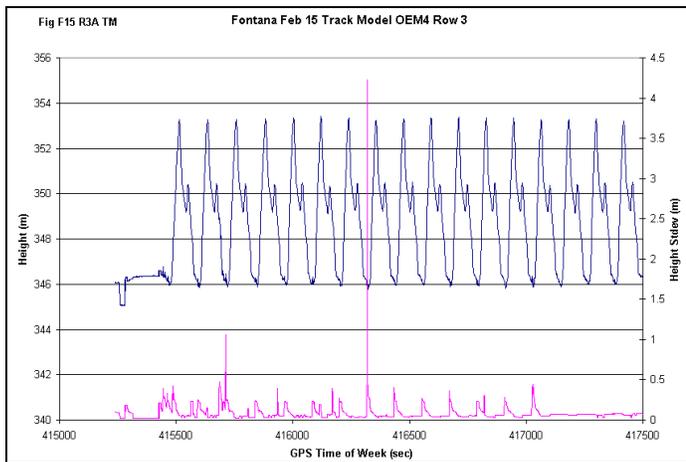
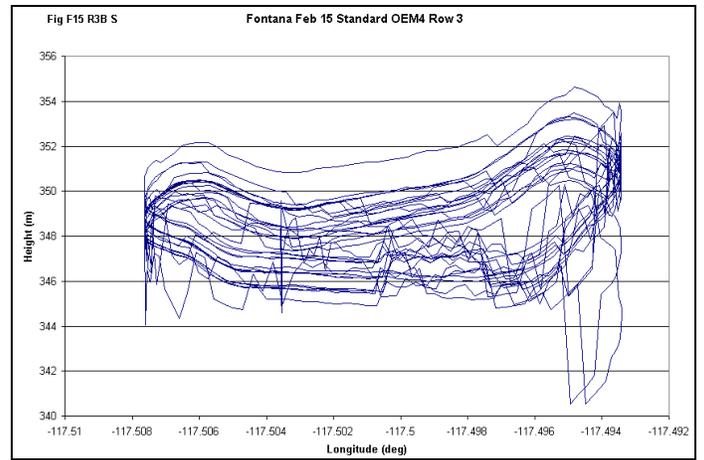
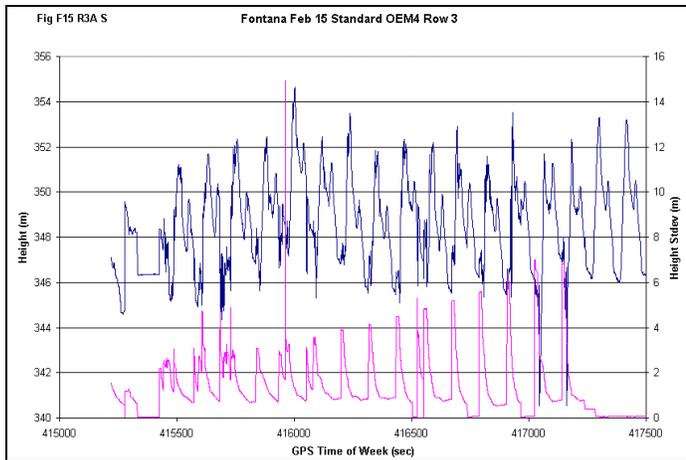
The above plots “Fig F15 R1D S” and “Fig F15 R1D TM” show the standard deviations for the standard OEM4 and track model OEM4 respectively, and a comparison of these with the associated error plots indicates that the reported standard deviation is a reasonable indication of the true accuracy of the system in these cases. The plots above at least show the assumption that the standard deviation truthfully reflects to error level is somewhat reasonable.

For every epoch of the data set related to the above test in which the system did not have fixed ambiguities, the ratio of the each axis error to its associated standard deviation was computed. Then the means of all these ratios for every axis and both systems were generated. These mean ratios are summarized below:

Table 2: Mean Ratio Axis Error and Standard Dev

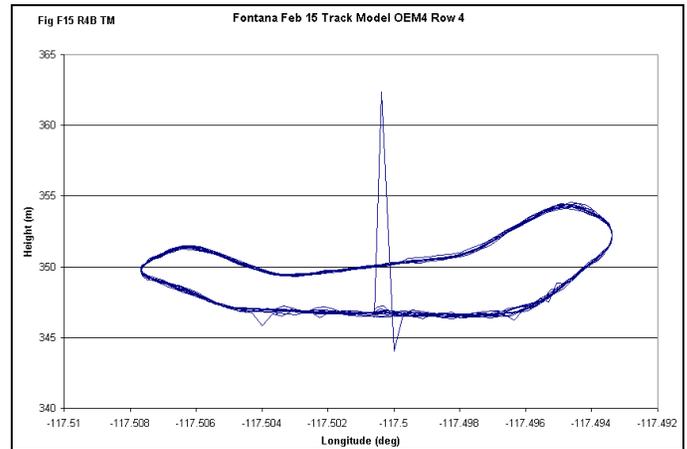
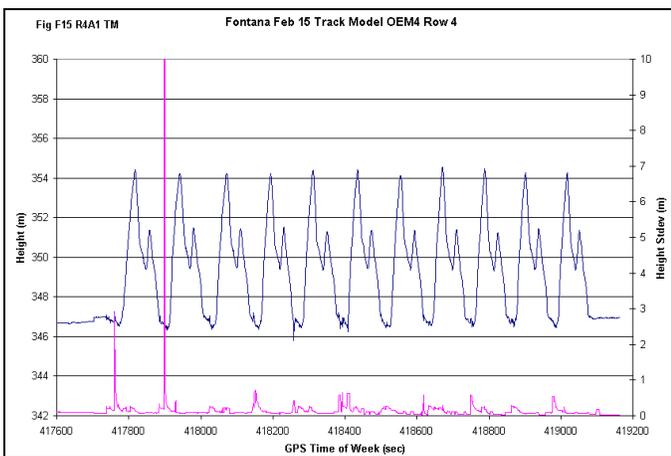
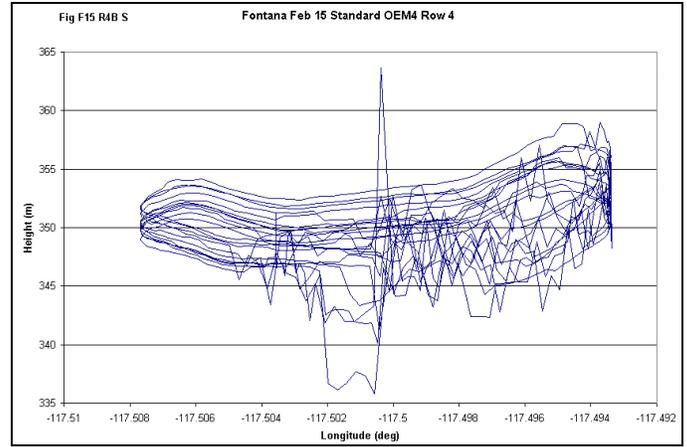
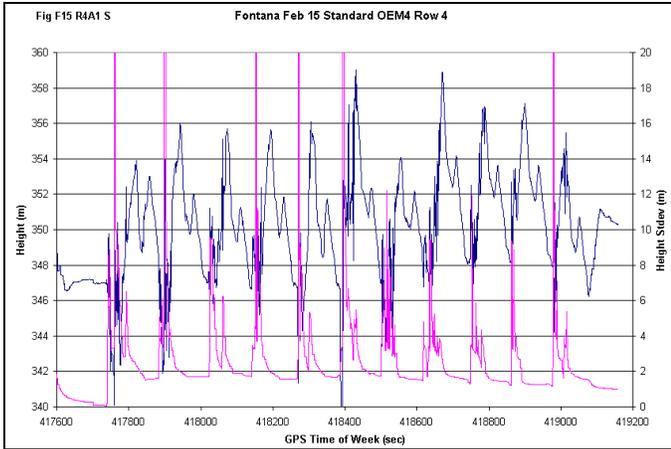
System	North	East	Height
Standard	0.66	0.64	0.52
Track Model	0.58	0.68	0.74

This also indicates that the standard deviations are a fair representation of the actual errors in both systems.



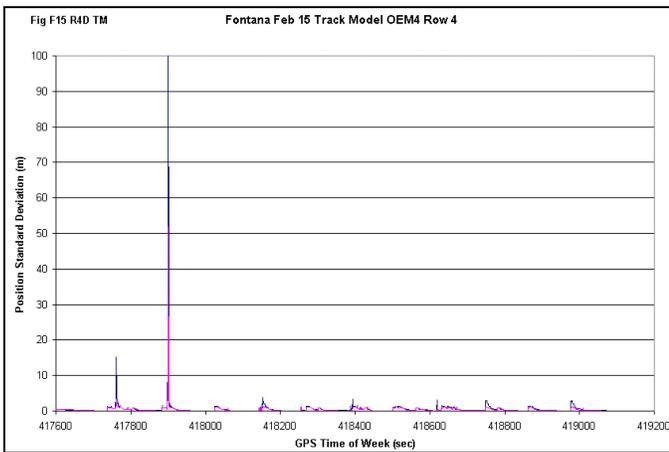
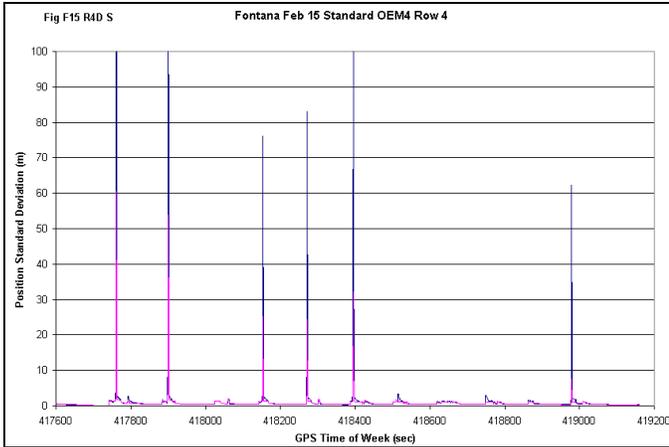
The plots “Fig F15 R3A S” and “Fig F15 R3A TM” above show the height repeatability over time and the height standard deviations over time for the two systems on Row 3. Row 3 is 6.6 metres away from the overhanging fence, and so there were enough natural obstacles so the systems normally had RTK resets at least once per lap. In the standard version, the heights are clearly more erratic. The standard deviations for that version indicate height errors that vary between 1 and 7 metres. The standard version managed 4 fixed ambiguity resolutions, and the constrained version was able to resolve 18 times.

The plots “Fig F15 R3B S” and “Fig F15 R3B TM” above show the height vs longitude for the standard (top) and constrained versions. The constrained version has much better repeatability, and although no control was available except the repeated track, it is clear there is a significant qualitative difference between the two systems. There is a 1.0 metre excursion at -117.5 on the constrained height plot. This is reflected as a 4.5 metre standard deviation.



The row 4 plots of height vs time are shown above on “Fig F15 R4A S” and “Fig F15 R4A TM”. The A and B plots show extreme variability for the standard model and very good repeatability for the track model OEM4 heights.

The plots Fig F15 R4B S, Fig F15 R4B TM seen above show the height vs longitude for the row 4 trajectory. The excursion seen at about -117.5 of the plot of the constrained system occurs because the horizontal position computed with poor geometry was outside the boundaries of all the planar sections of the track model, so no constraint could be found for it. The standard deviation of the height at this point is about 300 metres for both systems. This error can be eliminated simply by extending the planar section boundaries past the fence.



The plots “Fig F15 R4D S” and “Fig F15 R4D TM” seen above show the horizontal standard deviations for the standard and the constrained system. When the track model constraints act on the constrained system the horizontal standard deviations are almost always less than 3 metres, while the standard system has standard deviations which reach over 50 metres. So it appears that the track model is a significant help with the horizontal positions when the geometry is poor. This is partly because knowledge of height makes the other components more observable, but also because the tilt of the planar sections makes a portion of the horizontal position directly observable via the constraint. This is interesting because as the vehicle moves around the track which is composed of inwardly pointing planar sections, all of the position components are at one time or another directly observable by the planar section constraints. Provided the system can maintain carrier tracking on a minimal number of satellites, the accuracy improvement provided at one portion of the track can be carried forward to another portion of the track over which a different position component becomes observable. In this way eventually (say after 1 lap), all the position components can become known.

During the portions of the test that did not incorporate operator induced reset, some estimate of the expected operational accuracy can be made. This is based on the computed standard deviations and the assumption that the standard deviations computed actually do represent the errors in the system. This assumption is a reasonable one given the results of the controlled reset test carried out on the data from row 1 collected Feb 15 in which the standard deviations are a fair reflection of the measured error levels on both the standard OEM4 and track model OEM4 systems (see Table 2).

In order to quantify the error level of single axis positions, all the single axis standard deviations were put into 0 to 0.5, greater than 0.5 and greater than 1.0 m categories and the total for each category was generated. Since data was collected from different rows for different lengths of times, the accumulations are normalised as if each row collected 1000 seconds of data. The following tables summarise the results:

Table 4: Normalised Count of Errors Greater than 1 m

Model Axis	Row 1	Row 2	Row 3	Row 4
Std N	29	0	139	199
Std E	0	0	155	210
Std U	34	0	449	900
TM N	11	0	82	109
TM E	0	0	54	75
TM U	0	0	1	2

Table 5: Normalised Count of Errors Greater than .5 m

Model Axis	Row 1	Row 2	Row 3	Row 4
Std N	70	0	342	391
Std E	24	0	357	390
Std U	105	0	449	922
TM N	55	0	163	248
TM E	23	0	148	243
TM U	0	0	1	21

Table 6: Summary Reported Error Distribution

Model Axis	All Rows 4000 pts # greater than 0.5	Require Only 32% >0.5	All Rows 4000 pts # greater than 1.0	Require Only 5% >1.0
Std N	803	20.1	367	9.1
Std E	771	19.3	365	9.1
Std U	1476	36.9	1383	34.5
TM N	466	11.6	202	5.1
TM E	414	10.3	129	3.2
TM U	22	0.6	3	0.1

The results show that the standard model fails to meet the requirements of 1.0 m 95% of the time and at 0.5 m 68% of

the time. The OEM4 with supplementary track model constraints has no difficulty with the 68% 0.5 m requirement and is within acceptable limits for the 95% 1.0 m requirement.

CONCLUSIONS

From the test it is evident that the track model constraints improve the positioning accuracy significantly, up to a factor of 10 in many cases and sometimes more. In most cases, the improvement is in height as one would expect, but in conditions of poor geometry the horizontal accuracy is also much better (sometimes more than 100 times better) in the constrained case. The horizontal accuracy also improves depending on the slope of the constraining section with respect to the local level because if there is a significant slope, then a component of the planar section's normal vector will be parallel to the local level plane.

The track model could be extended (extrapolated) outside the ribbon of the track so that bad geometry cases such as the one seen on row 4 also have the use of a planar constraint.

The implementation on the OEM4 receiver is correct. There were a total of 11400 seconds of data, and less than 10 with height position errors of 1 metre or more.

The reported standard deviations accurately reflect the position errors in the system.

The idle time seen on the track model OEM4 is not significantly worse than on the standard OEM4.

The base station can use MSL control as long as the adjustment process is carried out and applied to the base station co-ordinates. Note that the adjustment doesn't account for the total undulation, all the computations in the filter are performed in ellipsoidal or ECEF co-ordinates anyway. The base station shift will account for undulation variations between the track model and the base station.

The base station shift is crucial to the successful operation of the track model constraints.

The track model constraints only work in the cases where there are at least 4 satellites. This isn't a big a problem because of the 4600 samples take at row 4, there were only 8 seconds of less than 4 satellite coverage.

The track model constraints make it possible for the system to provide 1.0 m accuracy at the 2 sigma level (95%) or 0.5 m accuracy at the 1 sigma (68%) level in a restricted environment such as the Fontana racetrack.

ACKNOWLEDGEMENTS

The authors would like to thank the following people without whose efforts the project could not have succeeded: Mike Bobye, Ian Williamson, Kim Deimert, Jason Hamilton, Jason Jones, Pat Fenton of NovAtel Inc., Michael Brown and Steve Lieber of Steve Lieber and Associates. I would also like to thank Janet Neumann for her review of the paper. Finally I would like to thank Grover Brown for general reference [3].

REFERENCES

- [1] Ford, T.J. and J. Neumann , NovAtel's RT20 - "A Real Time Floating Ambiguity Positioning System", Proceedings of ION GPS '94, Salt Lake City, Utah, Sept. 20-23, 1994, The Institute of Navigation, Washington, D.C. pp. 1067.
- [2] J. Neumann, A. Manz, T. Ford, O. Mulyk, "Test Results from a New 2 cm Real Time Kinematic GPS Positioning System", Proceedings of ION GPS '96, Kansas City, Mo., Sept. 17-20, 1996, The Institute of Navigation, Washington, D.C. pp. 183
- [3] R.G. Brown, P.Y.C Hwang, **Introduction to Random Signals and Applied Kalman Filtering** 3rd edition, John Wiley and Sons, 1997