Implementation and Experimental Validation of Cycle Ambiguity Resolution with Position Domain Integrity Risk Constraints

Samer Khanafseh, Steven Langel and Boris Pervan
Illinois Institute of Technology, Chicago, IL

ABSTRACT

Cycle ambiguity resolution in applications that require both high accuracy and high integrity is challenging. This paper describes new results, implementation details, and experimental validation for the Enforced Position-domain Integrity-risk Cycle resolution algorithm (EPIC). EPIC computes tight bounds on the integrity risk of cycle resolution by evaluating the impact of incorrect fixes in the position domain. In previous work, the fundamentals of the EPIC fixing method were described, mathematically formulated, and the algorithm's performance was quantified for simple snapshot-fixing systems that use least squares estimation techniques. A significant gain in navigation performance using EPIC was demonstrated when compared with other existing methods for constraining integrity risk. In this paper, EPIC's capabilities are expanded to more general applications that include Kalman filtering in the position estimation process. Practical implementation issues, such as using partially-fixed ambiguity vectors in the filter, dealing with satellite geometry changes and cycle slips, re-fixing the re-acquired satellites without jeopardizing integrity, and sensitivity analysis to the input parameters are also thoroughly addressed. Finally, experimental validation is provided and the results are compared with other methods for integrity-constrained cycle resolution.

INTRODUCTION

Accuracy and integrity risk are fundamental performance measures that affect navigation system availability. In Global Navigation Satellite Systems (GNSS), the integrity risk is quantified as the probability that the position error exceeds predefined alert limits. In order to provide GPS navigation for applications requiring high accuracy, carrier phase measurements are usually used. However, to extract high accuracy from carrier phase measurements, cycle ambiguities must be resolved. In navigation systems that have mild integrity risk requirements (such as geodetic survey or agricultural applications), all ambiguities can be fixed without paying much attention to the reliability of cycle resolution. For life-critical GNSS applications (such as civil aircraft landing with a Ground Based Augmentation System), extremely high levels of integrity are required [1]. If the integrity risk requirement is extremely stringent, the user can opt to ensure integrity, at the expense of the position solution's accuracy, by not fixing the ambiguities. With the emergence of new aviation applications such as autonomous airborne refueling [2], [3] and autonomous shipboard landing [4], [5], where the life of pilots is involved and the vehicles are highly dynamic, high levels of both integrity and accuracy are required simultaneously. In these situations, using carrier phase measurements to meet both the accuracy and integrity risk requirements can be very challenging.

Over the last two decades, a multitude of research has been conducted in the area of cycle resolution, and many different approaches have been developed to fix cycle ambiguities. A summary of the most frequently used methods is provided in [6] and [7]. To verify that a carrier phase GPS navigation system can provide adequate availability with the required high accuracy and integrity, it is necessary to establish whether or not the integrity risk of cycle resolution is sufficiently low prior to the intended operation. Some of these methods, such as integer bootstrapping [8], provide an *a priori* measure of the probability of fixing the ambiguities to their true integer values. The information from these methods can be used to determine the probability of incorrect fixing, which must be smaller than the integrity risk requirement. This means that whenever the *a priori* probability of incorrect cycle resolution exceeds the allocated integrity risk requirement, the operation is deemed unavailable.

In a method that has been used previously to account for the integrity risk of cycle resolution [9], a threshold on the Probability of Incorrect Fix (P_{IF}) is defined. The P_{IF}

threshold must be allocated from, and be smaller than, the allowable overall navigation integrity risk. During a typical navigation operation, only a subset of ambiguities (or linear combinations thereof) can be fixed with a joint incorrect fix probability lower than the threshold P_{IF}. The rest must be left as real-valued (floating) numbers. This set of partially fixed ambiguities is then used in position estimation and to generate the corresponding position domain protection levels. A protection level is a statistical bound for position error consistent with the allowable overall integrity risk; it is compared to a pre-defined position domain alert limit for the specific navigation operation. In the approach in [9] it is implicitly and conservatively assumed that all incorrect integer ambiguity candidates cause the position estimate errors to exceed the alert limit. Unfortunately, fixing a sufficient number of ambiguities to meet tight accuracy requirements with a success rate that meets stringent integrity risk requirement is not always possible [10] and [11]. In response, the Enforced Position-domain Integrity-risk Cycle resolution algorithm (EPIC) has been developed in [12].

EPIC is a new theoretical approach developed to quantify the position-domain integrity risk for cycle ambiguity resolution in satellite-based navigation systems. While the assumption that any incorrectly fixed cycle ambiguity vector causes hazardously large position errors is simple and practical, it is overly-conservative and can unnecessarily limit navigation availability for applications with stringent requirements for accuracy and integrity. By directly evaluating the impact of incorrect fixes in the position domain, EPIC computes tighter bounds on the navigation integrity risk resulting from cycle resolution. To accomplish this, an integer space is considered including all candidate sets of cycle ambiguities that cause the position error to fall within the alert limit boundaries. If a candidate set is incorrect, it will introduce biases in the estimated position error. As long as the sum of probabilities of the biased estimates that cause the position error to exceed the alert limit is lower than the integrity risk requirement, navigation integrity is preserved. However, due to the existence of these position domain biases, the practical use of this method has thus far been limited to simple snapshot-fixing systems that use least squares estimation techniques to estimate the position after fixing.

In this paper, we start with a brief overview of the different methods that have been used to verify the integrity risk of cycle resolution. We will then review the fundamental theoretical background for the EPIC algorithm that will serve as a basis for further derivations. Next, EPIC is extended to implementations in more general navigation systems that involve Kalman filters (such as inertial integrated GPS systems). To achieve this goal, we derive closed form expressions for the time varying position domain integrity risk at every epoch. Furthermore, to include and recover satellites that are newly-acquired after the initial fix (or re-acquired after a brief loss of lock) without jeopardizing the system's integrity, a 'Dual-Track' (DT) fixing method is introduced. In this method, an independent fixing track is implemented in which fixing of newly acquired satellites is continuously attempted while accounting for the integrity constraint. The DT–EPIC algorithm is then implemented in a Kalman filter, and its performance is demonstrated in scenarios where satellite blockages and outages are frequent and abrupt—such as in urban environments, Autonomous Airborne Refueling (AAR) and shipboard landing applications with steep bank angles. Finally, EPIC and DT–EPIC are experimentally validated by post-processing measurements collected at the Illinois Institute of Technology. Issues such as computational burden, sensitivity to input parameters, and real-time applicability are also thoroughly discussed.

INTEGRITY OF CYCLE RESOLUTION

Different methods have been developed to address the issue of the integrity of cycle resolution. Although certain methods, such as the ratio test, provide a measure of the quality of cycle resolution, it is not yet clear how to tie such a test to the integrity risk requirements [7] and [13]. Other methods such as [14] provide a probabilistic measure of the ambiguity resolution that can be used for integrity verification. However, this method provides an *a posteriori* measure and depends on access to the measurements. The bootstrap method, on the other hand, provides an *a priori* measure of the probability of correct cycle ambiguity estimation. Furthermore, since this probability was derived stochastically, it is easily implementable in applications with integrity risk requirements. Therefore, in this section we briefly describe several approaches to address the integrity of cycle resolution using the bootstrap fixing method.

CONVENTIONAL METHODS

This section briefly describes the method derived in [9], which we refer to here as the "conventional method". It provides a formula for the relation between the P_{IF} threshold, the fault-free integrity risk requirement (I_{H0req}) and the vertical protection level. To derive such a formula, it is easiest to start with the concept of the fault free Vertical Protection Level (VPL_{H0}). VPL_{H0} is defined in (1) as a statistical overbound such that the probability of the vertical position estimate error ($\left|\hat{x}_v - x_v\right|$) exceeding VPL_{H0} equals $I_{H0\,req}$.

$$P\left\{\left|\hat{x}_v - x_v\right| > VPL_{H0}\right\} = I_{H0req} \tag{1}$$

where,

P{■}: the probability of event '■'

$\hat{x}_v$: vertical component of the estimated relative position vector

x_v: vertical component of the true relative position vector

After fixing the ambiguities, two mutually exclusive and exhaustive events can be defined in relation to (1): a *correct fix* (CF) and an *incorrect fix* (IF) event. An IF includes all events where some or all ambiguities have been fixed incorrectly. A CF, on the other hand, includes only those events where no ambiguities are fixed incorrectly. Note that by definition of a correct fix, if we choose not to fix any ambiguity, thereby leaving the cycle ambiguities floating, then the Probability of Correct Fix (P_{CF}) is equal to 1. At the same time, if we choose to fix some or all ambiguities, we expect P_{CF} to be less than 1. Using the law of total probability, equation (1) can be expanded to,

$$\begin{aligned} P\left\{\left|\hat{x}_v - x_v\right| > VPL_{H0}\right\} = \\ P\left\{\left|\hat{x}_v - x_v\right| > VPL_{H0} \middle| CF\right\} P_{CF} \\ + P\left\{\left|\hat{x}_v - x_v\right| > VPL_{H0} \middle| IF\right\} P_{IF} \end{aligned} \quad (2)$$

In this method, the probability that the vertical error exceeds VPL_{H0} given that the cycle ambiguities are fixed incorrectly (i.e., $P\left\{\left|\hat{x}_v - x_v\right| > VPL_{H0} \middle| IF\right\}$) is conservatively assumed to be equal to 1. This assumption is considered conservative because in reality there might be incorrect fix events that result in a small vertical error such that $P\left\{\left|\hat{x}_v - x_v\right| > VPL_{H0} \middle| IF\right\}$ is less than 1.

Since the correct fix and incorrect fix events are mutually exclusive and exhaustive, as defined earlier, P_{CF} = 1 – P_{IF}. Using the conservative assumption $P\left\{\left|\hat{x}_v - x_v\right| > VPL_{H0} \middle| IF\right\} = 1$ in (2) and substituting the result in (1),

$$\begin{aligned} I_{H0req} &= P\left\{\left|\hat{x}_v - x_v\right| > VPL_{H0} \middle| CF\right\} P_{CF} + P_{IF} \\ &\Rightarrow P\left\{\left|\hat{x}_v - x_v\right| > VPL_{H0} \middle| CF\right\} = \frac{I_{H0req} - P_{IF}}{1 - P_{IF}} \end{aligned} \quad (3)$$

Under the assumption that the GPS measurement noise is bounded by a zero-mean Gaussian distribution, the vertical position error after fixing the cycle ambiguities correctly can be assumed to be a random variable with a Gaussian distribution of zero mean and standard deviation $\sigma_{v|CF}$. Therefore, a probability multiplier ($K_{VPL|CF}$) is calculated using the inverse of the cumulative normal distribution function of $P\left\{\left|\hat{x}_v - x_v\right| > VPL_{H0} \middle| CF\right\}$. For example, for $I_{H0\,req} = 10^{-7}$, in [9] a reasonable threshold for P_{IF} was found to be 10^{-8}, which results in $K_{VPL|CF}$ = 5.35. During a typical navigation operation, only a subset of ambiguities (or linear combinations thereof) can be fixed with a joint incorrect fix probability lower than the threshold P_{IF}. The rest of the ambiguities must be left as floating numbers. This set of partially fixed ambiguities is then used in position estimation and to generate $\sigma_{v|CF}$. VPL_{H0} is then calculated by multiplying $\sigma_{v|CF}$ by 5.35. This VPL_{H0} must comply with the maximum tolerable vertical error, which is known as the Vertical Alert Limit (*VAL*).

Using this method, fixing a sufficient number of ambiguities to meet the tight accuracy requirement with a success rate that meets the integrity risk requirement is not always possible [10] and [11]. Therefore, if a tight accuracy and integrity requirements exist, as we will encounter later, this method is not sufficient to provide reasonable navigation availability. One venue to improve the VPL_{H0} calculation is the conservative assumption, made after (2), that all incorrect integer candidates will cause the position errors to exceed the alert limits. This assumption was eliminated in the development of EPIC [12].

EPIC ALGORITHM

Instead of using the conservative assumption that the vertical error exceeds VPL_{H0} if ambiguities are fixed incorrectly, EPIC considers an integer space that includes all candidate sets of cycle ambiguities that cause the position error to fall inside the alert limit boundaries. This includes all sets, correct or incorrect fixes, as long as they cause the vertical error to be bounded by the alert limits with the allowable level of integrity risk. In other words, if a candidate set is incorrect but satisfies the position domain alert limit constraints, we recognize that it does not violate integrity. In the remainder of this section, we briefly explain the EPIC algorithm. For more details, see [12].

Although VPL can be computed in EPIC [12], it is easier to derive it and implement it by computing integrity risk directly. Instead of starting from the definition of VPL_{H0} that meets the integrity requirements, EPIC calculates the integrity risk (I_{H0}) induced by the cycle ambiguity candidates and compares it to the required fault free integrity risk (I_{H0req}). If the calculated integrity risk (I_{H0}) is less than the required integrity risk then the navigation system is considered available under fault-free conditions. Fault free integrity risk is defined as the probability that the vertical error exceeds *VAL* as shown in (4).

$$I_{H0} = P\left\{\left|\hat{x}_v - x_v\right| > VAL\right\} \quad (4)$$

Considering the mutually exclusive and exhaustive events *CF* and all possible incorrect fix events IF_k and using the law of total probability as shown in (2), equation (4) can be rewritten as,

$$\begin{aligned} I_{H0} &= P\left\{\left|\hat{x}_v - x_v\right| > VAL \mid CF\right\}P_{CF} \\ &\quad + \sum_{k=1}^{\infty} P\left\{\left|\hat{x}_v - x_v\right| > VAL \mid IF_k\right\}P_{IFk} \end{aligned} \tag{5}$$

where,

IF_k: event corresponding to the k^{th} incorrectly fixed cycle ambiguity vector

P_{IFk}: probability of occurrence of the k^{th} incorrect fix

Since it is impractical to calculate the series with an infinite number of incorrect cycle ambiguity vector candidates, the series is broken into two subseries: one represents what will later be the candidates of interest (k = 1→ l) and another which includes all remaining candidates ($k = l + 1$→ ∞) as shown in (7). Next, in order to avoid calculating the latter subseries the conservative assumption $P\left\{\left|\hat{x}_v - x_v\right| > VAL \mid IF_k\right\} = 1$ for $k = l + 1$→∞ is used. Using the fact that all incorrect fix events (including both the tested and untested candidates) are mutually exclusive and exhaustive and then simplifying and rearranging the resultant equation yields [12]

$$\begin{aligned} I_{H0} = \\ & 1 - \left(1 - P\left\{\left|\hat{x}_v - x_v\right| > VAL \mid CF\right\}\right)P_{CF} \\ & - \sum_{k=1}^{l}\left(1 - P\left\{\left|\hat{x}_v - x_v\right| > VAL \mid IF_k\right\}\right)P_{IFk} \end{aligned} \tag{6}$$

In [12], it has also been shown that similar expressions to Equation (6) can be written for requirements other than *VAL*, such as *LAL*, vertical accuracy (A_{ccv}) or lateral accuracy (A_{ccL}).

Equation (6) is a closed-form expression in which the effect of a set of ambiguity candidates on the position domain integrity risk is explicitly defined. In [12] it has been shown that if the series term in (6) (which represents the incorrect fix candidates to be tested) is neglected, the remaining expression yields an equivalent representation to (3) with *VAL* and I_{H0} replacing VPL_{H0} and I_{H0req}, respectively. Therefore, as long as the sum of the series in (6) is greater than zero (which is always true), the integrity risk computed using (6) will be lower than that given by the conventional method. Therefore, the EPIC algorithm portrayed in equation (6) provides a tighter bound on integrity risk than the conventional method. This in turn results in improved navigation availability as shown in [12].

An efficient method that is based on a sequential elimination procedure has been developed and detailed in [12] to construct the best cycle ambiguity candidates. In that work, it has been shown that its efficiency can be greatly enhanced and its computation time can be considerably reduced if the ambiguities to be fixed are decorrelated, for example using the Least-squares AMBiguity Decorrelation Adjustment (LAMBDA) method [15] or the Hassibi et al. decorrelation method that was based on the Lenstra, Lenstra and Lovasz (LLL) algorithm [16]. In this work, the derivations will be carried out under the assumption that the decorrelation method developed in LAMBDA is used. However, both EPIC and the derivation methodology in this work are still applicable if the LAMBDA decorrelation scheme is not used. For compactness of notation, we will refer to $P\left\{\left|\hat{x}_v - x_v\right| > VAL \mid CF\right\}$ and $P\left\{\left|\hat{x}_v - x_v\right| > VAL \mid IF_k\right\}$ as P_{VALCF}, and P_{VALIFk}, respectively. The details of how to compute P_{CF}, P_{IFk}, P_{VALCF}, and P_{VALIFk} have been provided in [12]. Therefore, these aspects of EPIC will be omitted in this work except for P_{VALCF}, and P_{VALIFk}, which we revisit in detail in the following paragraphs because of its relevance.

Calculating the probability that a set of ambiguities is the correct one depends on the fixing method utilized. In this work, the bootstrap method [8] is used for cycle resolution because it provides a closed form *a priori* probability mass function of the integer estimation error. The bootstrap rounding method fixes ambiguities sequentially and provides a measure of the P_{CF} at each step of the fixing process. The sequential adjustment is performed according to the cycle ambiguity conditional variances with the ambiguity having the lowest conditional variance being fixed first. The i^{th} conditional variance ($\sigma_{i|I}^2$), defined as the variance of the ambiguity i conditioned on the previous ambiguities in the set $I = \{1,2,\ldots,i\text{-}1\}$ being fixed, is the (i,i) element of the diagonal matrix $\mathbf{D}$ resulting from the $\mathbf{LDL}^{\mathrm{T}}$ decomposition of the decorrelated floating cycle ambiguity estimate error covariance matrix. At each step in this sequence (the m^{th} step for example), the probability of the bootstrapped integer estimate ($\breve{\mathbf{a}}$) (which is an m×1 vector) being any arbitrary integer candidate ($\mathbf{z}$) given that the correct fixed ambiguity is ($\mathbf{a}$) is given in [8] as,

$$P(\breve{\mathbf{a}} = \mathbf{z}) = \prod_{i=1}^{m}\left[\Phi\left(\frac{1 - 2\mathbf{l}_i^T(\mathbf{a} - \mathbf{z})}{2\sigma_{i|I}}\right) + \Phi\left(\frac{1 + 2\mathbf{l}_i^T(\mathbf{a} - \mathbf{z})}{2\sigma_{i|I}}\right) - 1\right] \tag{7}$$

where,

$\mathbf{l}_i$: the ith column vector of the unit lower triangular matrix ($\mathbf{L}^{-\mathrm{T}}$) resulting from $\mathbf{LDL}^{\mathrm{T}}$ decomposition of the ambiguity covariance matrix

m: the number of cycle ambiguities fixed $m \leq n$

$$\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}v^2\right) dv$$

Therefore, if we set **z** equal to **a** (the correct cycle ambiguity vector), Equation (7) produces

$$P(\breve{\mathbf{a}} = \mathbf{a}) = \prod_{i=1}^{m}\left[2\Phi\left(\frac{1}{2\sigma_{i|I}}\right) - 1\right] \tag{8}$$

which can be used to calculate P_{CF}. For P_{IF}, the value of (**a** - **z**) can be computed for the incorrect fix candidate of interest. For example, if a one cycle error on the first ambiguity is to be used as the first incorrect fix event (IF_1), then the vector $[1\ 0 \ldots 0]^{\mathrm{T}}$ is used for (**a** - **z**) in calculating P_{IF1}.

Before estimating the probability of the vertical error exceeding *VAL*, for simplification in notation, the conditional events *CF* and IF_k are replaced by a general event *B*. Later on, when we derive a methodology for estimating this probability, we will tackle the difference between the *CF* and *IF* events and discuss the specific approach for each one. By expanding the absolute value inside the conditional probability term of (6), the probability of the vertical error exceeding *VAL* given that event *B* took place can be written as:

$$\begin{aligned} P\left\{\left|\hat{x}_v - x_v\right| > VAL \mid B\right\} = {} & P\left\{\left(\hat{x}_v - x_v\right) < -VAL \mid B\right\} \\ & + P\left\{\left(\hat{x}_v - x_v\right) > VAL \mid B\right\} \end{aligned} \tag{9}$$

The position vector is linearly estimated using GPS measurements with errors that are assumed to be bounded by Gaussian distributions with zero mean. Therefore, the distribution of the vertical position estimate error will also be Gaussian with a standard deviation defined as $\sigma_{v|B}$. The mean of this distribution will depend on the event *B* and hence is referred to as μ_B. Knowing the mean (μ_B) and the standard deviation ($\sigma_{v|B}$), the probability of the first and second terms of the right hand side in (9) can be calculated using the normal cumulative distribution function at the limits -*VAL* and *VAL*, respectively.

Returning to the *CF* and IF_k events, the standard deviation of the vertical position error is not affected by an incorrect fix. Therefore, $\sigma_{v|B} = \sigma_{v|CF}$ for both the *CF* and IF_k events (the calculation of $\sigma_{v|CF}$ is detailed below). The only difference between these events is the mean of the Gaussian distribution. Since the position vector is estimated using unbiased estimators (such as least squares or Kalman filter estimators), the mean is zero ($\mu_{CF} = 0$) for the *CF* event (P_{VALCF}). In the case of IF_k events, the incorrect integer candidate will induce a bias in the relative position vector estimate. The resulting position domain bias caused by the k^{th} incorrect candidate $(\mathbf{a} - \mathbf{z})_k$ depends on the estimator used. Next, a method to compute $\sigma_{v|CF}$ and the bias for an example Kalman filter implementation is detailed. A similar procedure can be followed to derive $\sigma_{v|CF}$ and the bias for other estimators. P_{VALIFk} can then be calculated using a normal cumulative distribution function with standard deviation $\sigma = \sigma_{v|CF}$ and a mean (μ_{IFk}) equal to the vertical component of the position domain bias.

Assume that prior to the fixing step, the state estimate $\hat{\mathbf{S}}$, vector containing the floating cycle ambiguities and the relative position vector, and the associated estimate error covariance $\hat{\mathbf{P}}$ are expressed as, $\hat{\mathbf{S}} = \begin{bmatrix} \hat{\mathbf{x}}_{3\times1} \\ \hat{\mathbf{a}}_{n\times1} \end{bmatrix}$ and $\hat{\mathbf{P}} = \begin{bmatrix} \mathbf{P}_{\hat{\mathbf{x}}} & \mathbf{P}_{\hat{\mathbf{x}}\hat{\mathbf{a}}} \\ \mathbf{P}_{\hat{\mathbf{a}}\hat{\mathbf{x}}} & \mathbf{P}_{\hat{\mathbf{a}}} \end{bmatrix}$. If the LAMBDA decorrelation scheme is used, the i^{th} ambiguity in the Z-decorrelated space $\hat{a}_z^{(i)}$ is rounded to its nearest integer $a_z^{(i)}$.

$$a_z^{(i)} = \mathrm{round}\left(\begin{bmatrix}\mathbf{0}_{1\times3} & \mathbf{Z}_{i\ (1\times n)}\end{bmatrix}\breve{\mathbf{S}}^{(i-1)}\right) \tag{10}$$

where,

$a_z^{(i)}$: i^{th} rounded ambiguity element

$\mathbf{Z}_i$: i^{th} row of the transformation matrix **Z**. Notice, however, that if the LAMBDA decorrelation is not used, $\mathbf{Z} = \mathbf{I}$

$\breve{\mathbf{S}}^{(i-1)}$: updated state vector **S** based on fixing ambiguities 1 to *i* – 1, such that $\breve{\mathbf{S}}^{(0)} = \hat{\mathbf{S}} = \begin{bmatrix}\hat{\mathbf{x}}^T & \hat{\mathbf{a}}^T\end{bmatrix}^T$.

Next, the position vector estimate and remaining floating ambiguities $\breve{\mathbf{S}}$ are updated accordingly as follows:

$$\begin{aligned}\mathbf{K} = {} & \\ & \breve{\mathbf{P}}^{(i-1)}\begin{bmatrix}\mathbf{0}_{1\times3} & \mathbf{Z}_i\end{bmatrix}^T\left(\begin{bmatrix}\mathbf{0}_{1\times3} & \mathbf{Z}_i\end{bmatrix}\breve{\mathbf{P}}^{(i-1)}\begin{bmatrix}\mathbf{0}_{1\times3} & \mathbf{Z}_i\end{bmatrix}^T\right)^{-1}\end{aligned} \tag{11}$$

$$\breve{\mathbf{S}}^{(i)} = \breve{\mathbf{S}}^{(i-1)} + \mathbf{K}\left(N_z^{(i)} - \begin{bmatrix}\mathbf{0}_{1\times3} & \mathbf{Z}_i\end{bmatrix}\breve{\mathbf{S}}^{(i-1)}\right) \tag{12}$$

$$\breve{\mathbf{P}}^{(i)} = \left(\mathbf{I} - \mathbf{K}\begin{bmatrix}\mathbf{0}_{1\times3} & \mathbf{Z}_i\end{bmatrix}\right)\breve{\mathbf{P}}^{(i-1)} \tag{13}$$

where $\breve{\mathbf{P}}^{(i)}$ is the covariance of the updated state vector after fixing the i^{th} cycle ambiguity, such that $\breve{\mathbf{P}}^{(0)} = \hat{\mathbf{P}}$. This process is repeated for the desired number of fixed ambiguities (for example *m* times). At this point, $\sigma_{v|CF}$ can be computed as the square root of the (3, 3) element of $\mathbf{P}^{(m)}$, assuming that local east-north-up coordinate system is used for positioning.

$$\sigma_{v|CF} = \sqrt{\breve{\mathbf{P}}^{(m)}_{(3,3)}} \tag{14}$$

To calculate the position domain bias, we start by evaluating the impact on the state vector of fixing *m* ambiguities ($m \le n$), which is done by replacing Equations (11) and (12) by their equivalents:

$$\mathbf{K}_m = \hat{\mathbf{P}} \begin{bmatrix} \mathbf{0}_{m\times3} & \mathbf{Z}_{1:m\,(m\times n)} \end{bmatrix}^T \left(\begin{bmatrix} \mathbf{0}_{m\times3} & \mathbf{Z}_{1:m\,(m\times n)} \end{bmatrix} \hat{\mathbf{P}} \begin{bmatrix} \mathbf{0}_{m\times3} & \mathbf{Z}_{1:m\,(m\times n)} \end{bmatrix}^T \right)^{-1} \tag{15}$$

$$\breve{\mathbf{S}}^{(m)} = \hat{\mathbf{S}} + \mathbf{K}_m \left(\mathbf{a}_{z\,(m\times1)}^{(1:m)} - \begin{bmatrix} \mathbf{0}_{m\times3} & \mathbf{Z}_{1:m\,(m\times n)} \end{bmatrix} \hat{\mathbf{S}} \right) \tag{16}$$

where $\mathbf{Z}_{1:m}$ is made of rows 1 to *m* of the transformation matrix **Z**. If the LAMBDA decorrelation is not used, $\mathbf{Z}_{1:m}$ is replaced by $\mathbf{Z}_{1:m} = \begin{bmatrix} \mathbf{I}_{m\times m} & \mathbf{0}_{m\times(n-m)} \end{bmatrix}$.

Assume that the correct partially-fixed ambiguity vector is $\mathbf{a}_{(m\times1)}$ and the incorrect ambiguity vector is $\mathbf{z}_{k(m\times1)}$. The estimated states ($\mathbf{S}_{\mathrm{CF}}$) after fixing the ambiguities correctly and the states ($\mathbf{S}_{IF}$) after fixing the ambiguities incorrectly, respectively become:

$$\breve{\mathbf{S}}_{CF}^{(m)} = \hat{\mathbf{S}} + \mathbf{K}_m \left(\mathbf{a} - \begin{bmatrix} \mathbf{0}_{m\times3} & \mathbf{Z}_{1:m} \end{bmatrix} \hat{\mathbf{S}} \right) \tag{17}$$

$$\breve{\mathbf{S}}_{IFk}^{(m)} = \hat{\mathbf{S}} + \mathbf{K}_m \left(\mathbf{z}_k - \begin{bmatrix} \mathbf{0}_{m\times3} & \mathbf{Z}_{1:m} \end{bmatrix} \hat{\mathbf{S}} \right) \tag{18}$$

Therefore, the entire state vector difference ($\mathbf{b}_k$) caused by the k^{th} incorrect ambiguity candidate vector ($\mathbf{a} - \mathbf{z}_k$) is computed by subtracting (18) from (17):

$$\mathbf{b}_k = \breve{\mathbf{S}}_{CF}^{(m)} - \breve{\mathbf{S}}_{IFk}^{(m)} = \mathbf{K}_m \left(\mathbf{a} - \mathbf{z}_k \right) \tag{19}$$

The position domain bias is then extracted from the elements in $\mathbf{b}_k$ corresponding to the position states. Then, μ_{IFk} can be determined from the vertical component of the position domain bias.

Due to the existence of these position domain biases, the practical use of this method in [12] has been limited to simple snapshot-fixing systems that use least squares estimation techniques to estimate the relative position vector after fixing. In the next section, EPIC will be extended to more general navigation systems by laying out the fundamentals of computing the position domain integrity risk continuously over time. In these derivations we take as an example the Kalman filter implementation because of its abundant use as an estimator in various navigation applications. The same procedure can be followed to derive the integrity risk computation for other types of estimators or implementations.

KALMAN FILTER IMPLEMENTATION FOR EPIC

In the original EPIC algorithm, fixing ambiguities takes place only once to ease the computation of the integrity risk and to ensure that the risk does not grow overtime as additional ambiguities are fixed. In this section we define the necessary steps and derive the principal formulas for computing the integrity risk of the position estimate over consecutive Kalman filter updates. If the integrity risk in (6) is to be computed while updating the position estimates using a Kalman filter, the incorrect fix candidates that have been chosen at the fixing step, as well as P_{CF}, P_{IFk}, P_{VALCF} and P_{VALIFk}, must be reevaluated. Some of these quantities might change with time due to the nominal satellite geometry change or due to acquisition or loss of satellites. We will first discuss the nominal geometry change and then focus on acquiring and losing satellites. Since it is assumed that fixing the ambiguities takes place only once (snap-shot fixing), as long as the same ambiguities are maintained in the Kalman filter (either float, partially fixed or all-fixed), P_{CF} and P_{IFk} are not affected by satellite geometry change and remain the same. On the other hand, the standard deviation of the position estimate $\sigma_{v|CF}$ (after fixing the ambiguities) differs as the satellite geometry varies. In addition, since the state estimate vectors using the correctly and incorrectly fixed ambiguities change in time due to the change in geometry, $\mathbf{b}_k$ (and hence the position domain bias and μ_{IFk}) also will vary. Remember that both $\sigma_{v|CF}$ and $\mathbf{b}_k$ are used in the computation of P_{VALCF} and P_{VALIFk}.

As the satellite geometry changes, the estimator automatically provides the updated $\sigma_{v|CF}$. Therefore, we focus our attention on generating and updating the position domain bias over time. Remember that the position domain bias is extracted from the corresponding position state elements in $\mathbf{b}_k$, which by definition is the difference between the state vector estimated using the correctly fixed ambiguities and the one using the k^{th} incorrect ambiguity set:

$$\mathbf{b}_k = \breve{\mathbf{S}}_{CF} - \breve{\mathbf{S}}_{IFk} \tag{20}$$

After the fixing step, the state estimate is propagated forward through the Kalman filter time update and then through the measurement update. In the time update, from time $j - 1$ to time *j*, for example, the updated bias is defined as:

$$\bar{\mathbf{b}}_{k,j} = \bar{\breve{\mathbf{S}}}_{CF,j} - \bar{\breve{\mathbf{S}}}_{IFk,j} \tag{21}$$

where a crown on top of the state estimate vector **S** denotes the state vector after integer fixing has occurred and a bar on top denotes an *a priori* estimate (before using the measurements at that epoch *j*). However, both the correct and incorrect state estimate vectors are updated using the same dynamic model as:

$$\begin{aligned} \bar{\breve{\mathbf{S}}}_{CF,j} &= \mathbf{\Phi}_j \hat{\breve{\mathbf{S}}}_{CF,j-1} \\ \bar{\breve{\mathbf{S}}}_{IFk,j} &= \mathbf{\Phi}_j \hat{\breve{\mathbf{S}}}_{IFk,j-1} \end{aligned} \qquad (22)$$

where $\mathbf{\Phi}$ is the state transition matrix and a hat denotes the *a posteriori* estimate (after applying the measurements). Substituting (22) into (21) and using the definition of $\mathbf{b}_k$ from (20) for time $j - 1$ results in,

$$\bar{\mathbf{b}}_{k,j} = \mathbf{\Phi}_j \left(\hat{\breve{\mathbf{S}}}_{CF,j-1} - \hat{\breve{\mathbf{S}}}_{IFk,j-1} \right) = \mathbf{\Phi}_j \hat{\mathbf{b}}_{k,j-1} \qquad (23)$$

where the initial $\hat{\mathbf{b}}_{k,0}$ is $\mathbf{b}_k$ at the fixing step which is computed using Equation (19).

Equation (23) is used to update $\mathbf{b}_k$ as the state estimate vector is updated through a Kalman time update. At the measurement update, the state vector using the correct fix and incorrect fix ambiguities are updated by the measurement vector $\mathbf{y}_j$, the observation matrix $\mathbf{H}_j$ and the Kalman gain $\mathbf{K}_j$ as in Equations (24) and (25), respectively.

$$\hat{\breve{\mathbf{S}}}_{CF,j} = \bar{\breve{\mathbf{S}}}_{CF,j} + \mathbf{K}_j \left(\mathbf{y}_j - \mathbf{H}_j \bar{\breve{\mathbf{S}}}_{CF,j} \right) \qquad (24)$$

$$\hat{\breve{\mathbf{S}}}_{IFk,j} = \bar{\breve{\mathbf{S}}}_{IFk,j} + \mathbf{K}_j \left(\mathbf{y}_j - \mathbf{H}_j \bar{\breve{\mathbf{S}}}_{IFk,j} \right) \qquad (25)$$

Using the definition of $\mathbf{b}_k$ in Equation (20) and subtracting Equation (25) from Equation (24) results in:

$$\hat{\mathbf{b}}_{k,j} = \hat{\breve{\mathbf{S}}}_{CF,j} - \hat{\breve{\mathbf{S}}}_{IFk,j} = \left(\mathbf{I} - \mathbf{K}_j \mathbf{H}_j \right) \bar{\mathbf{b}}_{k,j} \qquad (26)$$

The satellite geometry change will be captured in the observation matrix $\mathbf{H}_j$ and the Kalman gain $\mathbf{K}_j$. Equations (23) and (26) provide the mechanism of projecting this variation to $\mathbf{b}_k$. As $\mathbf{b}_{k,j}$ is computed at every epoch, $\mu_{IFk,j}$ is extracted and used together with $\sigma_{v|CF,j}$ to compute $P_{VALCF,j}$ and $P_{VALIFk,j}$. These in turn are used in Equation (6) to compute the integrity risk at every epoch j.

If there is an abrupt change in the satellite geometry such as acquiring new satellites, losing satellites or cycle slips, further steps need to be taken. Typically, if a satellite sets or gets blocked, its corresponding states in $\mathbf{S}$ (associated ambiguities for example) are removed. Since these ambiguities have been removed, the states in $\mathbf{b}_k$ corresponding to that satellite must also be removed. Also, any incorrect ambiguity candidate set that has integer offsets for that specific satellite must be removed together with their associated probabilities (P_{IFk}). In contrast, it is quite challenging to adjust P_{CF} after taking those integers out of the solution. However, since the P_{CF} computation is the product of the individual conditional probabilities corresponding to the fixed ambiguities (7), taking integers out of the P_{CF} computation should make P_{CF} larger. Therefore, preserving the original P_{CF} from the fixing step when a satellite goes out is a conservative assumption.

If a satellite is acquired after the fixing step, it would be beneficial to use its measurement in the Kalman filter. Usually, the corresponding states for the new satellite are initialized before being used in the Kalman filter. Since the ambiguity state for that satellite is not fixed yet, the bias state associated with it should be initialized to zero. The process of fixing the new acquired satellites after an initial fixing step without jeopardizing integrity, is challenging. Hypothetically, the initial fixing step may have consumed most of the budget from the integrity risk requirement allocated for cycle resolution. It has already been shown that EPIC provides a tight bound on the integrity risk of cycle resolution, leaving very little margin for a new fixing attempt. In addition, recall that the bootstrap fixing method requires the computation of conditional variances in order to compute P_{CF}. These computations can be carried out provided that no prior fixing attempts have been conducted. Once a fixed solution has been achieved, any new fixing attempts will have to account for the initial fixing attempt in the computation of P_{CF}, which is not a trivial task. In light of these issues, newly acquired satellites are not fixed and therefore, P_{CF} and P_{IFk} initial values remain unchanged. In this work we adopt a new fixing algorithm that is able to initiate a new fixing attempt at any time without taking benefit of the already fixed solution. The only question that remains is whether or not a new fixing attempt should only be conducted at epochs where new satellites are acquired. The answer to this question in addition to the concept of the new fixing algorithm is discussed in the next section.

DUAL TRACK FIXING ALGORITHM

The basic premise of the dual track fixing algorithm is to continually attempt to fix a current floating satellite set subject to integrity risk requirement. Once the cycle ambiguities have been resolved, or partially resolved, any additional fixing might cause an integrity breach. One simple way to insure integrity while fixing newly acquired satellites is to perform an entirely new fix that does not rely on any previously fixed-integer information. In other words, if fixing is performed on a float solution, the whole integrity risk budget can be used again (but without taking credit for previous fixes). This new fixing attempt can be applied every epoch. In this way, we can ensure the use of the best available satellite geometry and the best fixed ambiguities (subject to integrity risk requirements) for positioning without jeopardizing integrity. This algorithm is shown schematically in Figure 1.

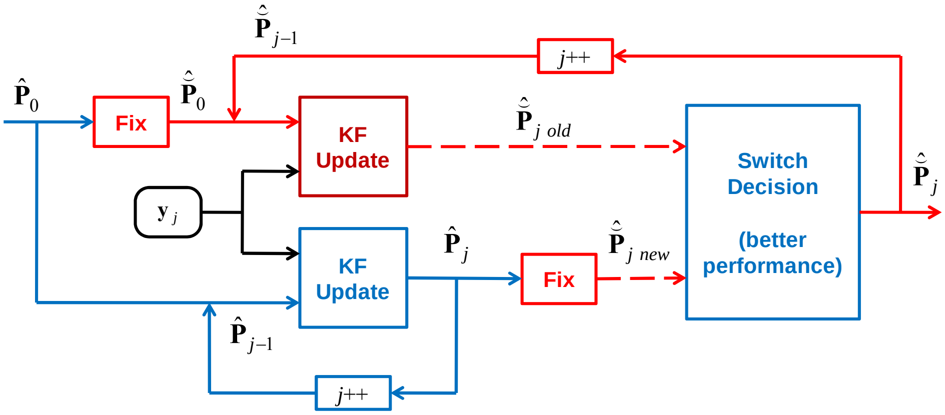
$\hat{\breve{\mathbf{P}}}_{j-1}$
j++
$\hat{\mathbf{P}}_0$
Fix
$\hat{\breve{\mathbf{P}}}_0$
KF Update
$\hat{\breve{\mathbf{P}}}_{j\ old}$
Switch Decision
(better performance)
$\hat{\breve{\mathbf{P}}}_j$
$\mathbf{y}_j$
KF Update
$\hat{\mathbf{P}}_j$
Fix
$\hat{\breve{\mathbf{P}}}_{j\ new}$
$\hat{\mathbf{P}}_{j-1}$
j++

Figure 1: Dual track fixing algorithm

The dual track fixing algorithm begins assuming that there is an initial floating solution with a state estimate vector $\hat{\mathbf{S}}_0$ and associated covariance matrix $\hat{\mathbf{P}}_0$. (Note that for clarity, only the covariance inputs and outputs are shown in Figure 1.) Integer fixing using EPIC is conducted next, which outputs a state vector with fixed ambiguities (maybe partially fixed) $\hat{\breve{\mathbf{S}}}_0$ and covariance matrix, $\hat{\breve{\mathbf{P}}}_0$. After the initialization step, there are two options. First, to use the current measurement ($\mathbf{y}_j$) at time *j* and propagate the previous solution forward using the Kalman filter propagation equations, which provides the current estimate vector $\hat{\breve{\mathbf{S}}}_{j\,old}$ and covariance matrix using the old-fixed ambiguities $\hat{\breve{\mathbf{P}}}_{j\,old}$. The EPIC integrity risk at this stage is computed using Equations (23) and (26). This option, shown as the upper track in Figure 1, is representative of how the original EPIC algorithm works [12].

The other option is to use a Kalman filter to propagate the float solution $\hat{\mathbf{S}}_{j-1}$ and $\hat{\mathbf{P}}_{j-1}$ forward. Then, a new fixing attempt is executed using EPIC. The output solution of the new fixing step is denoted as $\hat{\breve{\mathbf{S}}}_{j\,new}$ and $\hat{\breve{\mathbf{P}}}_{j\,new}$. Notice that in the second track, shown as the lower track in Figure 1, only the float estimates are used in this Kalman filter loop. Therefore, it is permissible to use the whole integrity risk requirement because it did not profit from the old fixing step. Since at the output of the two tracks we have two solutions, a decision must be made. If the EPIC integrity risk is smaller using the newly fixed solution (lower track) than using the old one (upper track), then $\hat{\breve{\mathbf{S}}}_{j\,new}$ and $\hat{\breve{\mathbf{P}}}_{j\,new}$ will replace $\hat{\breve{\mathbf{S}}}_j$ and $\hat{\breve{\mathbf{P}}}_j$, respectively, and will be used in the old fixed solution on the upper track from this point forward. If the EPIC solution for the newly fixed ambiguities is not better than the old one, then no switch is made and the new fixed solution $\hat{\breve{\mathbf{S}}}_{j\,new}$ and $\hat{\breve{\mathbf{P}}}_{j\,new}$ is discarded. The execution of this process at every epoch has three main advantages. First, the dual track algorithm provides a means of fixing newly acquired satellites or recovering satellites that have been momentarily blocked. Secondly, the increased filtering time achieved in the lower track can potentially improve observability on the floating cycle ambiguity estimates, resulting in a larger number of fixed ambiguities. Finally, accuracy and integrity risk requirements may be changing during the duration of a given operation. Hence, an initial fixed solution that meets the requirements at the current epoch may not satisfy the requirements at a future epoch, causing a loss of continuity and rendering the operation unavailable. The dual track algorithm can produce a better solution at a given epoch that complies with the current accuracy and integrity risk requirements.

In previous work [12], the performance of the EPIC algorithm was quantified for an autonomous shipboard landing application as an example. However, in that work, it was assumed that the landing was achieved using a straight-in approach, where satellite geometry change is minimal. In order to investigate the performance using the Dual Track (DT) fixing algorithm and EPIC in a Kalman filter implementation, a shipboard aircraft landing application using a curved approach is considered next.

PERFORMANCE QUANTIFICATION SIMULATIONS

In this section, we quantify the performance of EPIC (applied in a Kalman filter) with the dual track fixing algorithm (EPIC+DT) and consider for shipboard landing in the presence of satellite blockage. Aircraft carrier recovery patterns flown by Navy pilots are quite different depending on the time of day, prevailing weather conditions and visibility. For example, in adverse weather conditions and during all night flight operations the pilot conducts a case-III approach (straight-in approach). Due to the lack of visibility, case-III recoveries are conducted one aircraft at a time using TACtical Air Navigation system (TACAN). On the other hand, if the pilot has good visibility and is returning to the ship during the daytime, then he/she conducts a case-I approach, shown in Figure 2. The advantage of this flight pattern is that it allows the carrier to recover more aircraft during any given time period than possible using a Case-I approach pattern. Providing instrument landing support for Case-I approaches become necessary if unmanned aerial vehicles are incorporated into existing carrier fleets.

The Case-I recovery is a well established carrier landing approach that is routinely executed by Navy pilots and will serve as the mission profile for the simulations in this work. During a Case-I recovery the aircraft must complete a series of high banking maneuvers which will inevitably result in satellite blockage. This landing pattern is composed of six distinct phases. The pilot first enters a holding pattern (phase 1) until he/she receives notification from the ship that the aircraft is cleared to land. Upon receiving clearance, the pilot breaks out of the holding pattern and descends to 'initial'; the beginning of the final landing approach (phase 2). A constant altitude fly-by is then conducted (phase 3) followed by a 180 degree turn at a high bank angle (possibly exceeding 60^{o}) (phase 4). Finally, the aircraft proceeds back towards the carrier (phase 5) and completes another 180 degree turn with a bank angle of up to 30^{o} before arriving on the flight deck. It is important to realize that the banking during this approach ultimately results in serious satellite blockage.

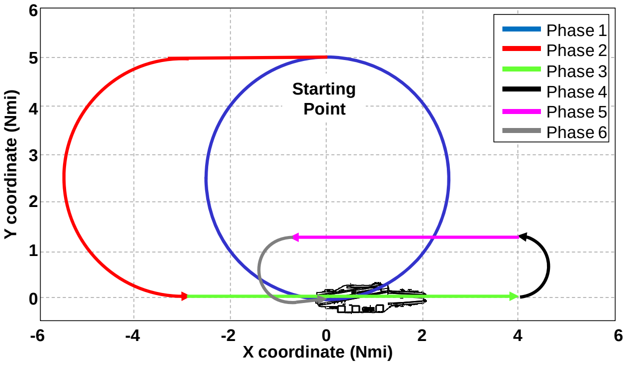

Figure 2: Case-I recovery profile (ship is not to-scale)

Let us now describe the navigation architecture in which EPIC will be implemented. This navigation architecture is designed to support a specific application: autonomous shipboard landing. Because of the mobility of the reference station in a shipboard-relative landing application, higher levels of accuracy are required than for similar precision approach applications at land-based airfields. In addition, to ensure safety and operational usefulness, the navigation architecture must provide high levels of integrity and availability. Because of the highly stringent requirements, the navigation system is based on Carrier Phase Differential GPS (CPDGPS) positioning. However in order to benefit from the high precision of CPDGPS the correct resolution of cycles must be ensured. A number of methods have been used in prior work to aid in the high-integrity cycle resolution. Satellite motion can provide the observability of the cycle ambiguities [17]. Unfortunately the rate of satellite motion is relatively slow in comparison with the time scales of the mission.

Heo, et al. have proposed a GPS navigation algorithm for autonomous shipboard landing applications where geometry-free/divergence-free code-carrier filtering is performed continuously for visible satellites on both the aircraft and the ship until the aircraft is close to the ship [5] [10]. Geometry-free filtering [18], by definition, does not depend on the geometry of the satellites or the user location and eliminates all error sources except for receiver noise and multipath. A geometry free measurement of the widelane cycle ambiguity is formed by subtracting the narrowlane pseudorange from the widelane carrier [17], [18]. A drawback of the geometry free measurement is the presence of higher noise relative to the L1 and L2 carrier phase measurements. This can be overcome by filtering the geometry free measurement over time prior to the final approach. In order to model colored multipath noise in the geometry free measurements, a first order Gauss-Markov measurement error model is used. In this work, a time constant of one minute for the ship and 30 seconds for aircraft is assumed. The outputs of the filtering process are the floating widelane cycle ambiguity estimates. When the aircraft is close to the ship, floating L1/L2 cycle ambiguity estimates can be extracted with the aid of satellite geometric redundancy [4], [5]. Next, the cycle ambiguities are fixed using the bootstrap method [8]. For the conventional method, the bootstrap rounding process is performed for those ambiguities that can be fixed with a P_{IF} that is lower than a threshold of 10^{-8} (for an overall fault-free integrity risk requirement of 10^{-7}, for example). The remaining ambiguities remain floating. With the ambiguities partially fixed, the standard deviation of the position estimate error ($\sigma_{v|CF}$) is used to calculate VPL_{H0}. Using EPIC, on the other hand, the position domain integrity of cycle resolution (I_{H0}) is calculated using (6).

The performance, therefore, can be predicted for a single approach through a covariance analysis and availability computation. To account for the GPS satellite geometry change, availability analysis is performed by simulating 1440 approaches (one approach per minute during the day). Because of the aircraft banking in the Case-I approach the navigation performance is sensitive to the ship heading. Based on [19], the worst performance is when the aircraft heading at touchdown (ship heading) is in the south direction. A given approach is said to be available if the accuracy and integrity requirements will be satisfied at each point along the approach. In the conventional method, availability is calculated as the percentage of approaches for which VPL_{H0} and $\sigma_{v|CF}$ during the entire approach are less than VAL and the accuracy requirement, respectively. Using EPIC, availability is calculated as the percentage of approaches for which I_{H0} and P_{acc} during the entire approach are less than the requirements ($I_{H0\ req}$ *and* $P_{acc\ req}$, respectively). For the autonomous shipboard landing application, accuracy and integrity requirements vary as functions of distance to touchdown. The logic behind the requirement variation is that these requirements will become more stringent as the aircraft comes closer to completing its mission. As we proceed, a hypothetical example of required vertical positioning performance will be defined as a function of distance to touchdown.

Three factors must be considered when determining where to first fix the cycle ambiguities: geometry-free pre-filtering duration, the dynamic nature of the performance drivers and robustness to ionospheric and tropospheric error models. Of course, prior to the initial fixing point, we wish to pre-filter as long as possible since this will improve the cycle ambiguity resolution. However note that before reaching the initial fix point, the navigation architecture is only providing floating solutions for positioning. Since the performance criteria tighten as the aircraft comes closer to landing, there exists a point where the floating solution will no longer satisfy the requirements, rendering the approach unavailable. Lastly, the Kalman filter propagation must be robust to the ionospheric and tropospheric decorrelation error models. If the baseline between the aircraft and ship becomes too large, the fidelity in the error models diminishes. In order to accommodate these issues, the fixing point shown in Figure 3 was selected. Based on [19], this point, in the 5th phase of the approach, is approximately 2 nautical miles away from the ship, and represents a good compromise for the issues described above.

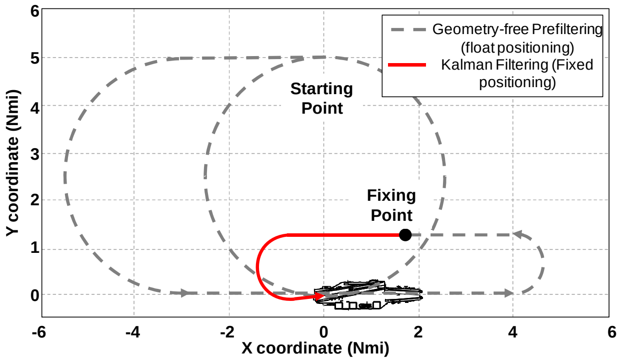

Figure 3: Initial fix point (ship is not to-scale)

The requirements and simulation parameters that are used in this work are based on those given in [5], [10] and [19]. Example requirements on VAL and accuracy (A_{cc}) as a function of the distance to touchdown along the flight path (d) are shown in Table I. In this simulation, the standard deviation of the carrier phase (single difference) measurement noise $\sigma_{\Delta\phi}$ is assumed to be 1 cm and the standard deviation of the pseudorange (single difference) measurement noise $\sigma_{\Delta PR}$ is assumed to be 50 cm. The single difference standard deviations $\sigma_{\Delta PR}$ and $\sigma_{\Delta\phi}$ are related to the raw values (σ_{PR} and σ_{ϕ}) by a scaling factor of $\sqrt{2}$. In this analysis, geometry-free prefiltering is assumed to start at the beginning of the approach and is used to generate floating estimates of the widelane cycle ambiguities. The rest of the simulation parameters are summarized in Table I. The performance of EPIC+DT algorithm is compared to both the EPIC-only and the conventional methods.

Table I Simulation Parameters

Parameter	Value
$I_{H0\ req}$	10^{-7}
$P_{acc\ req}$	0.05
VAL	$d \leq 0.5$ nmi : $VAL = 1.8$ m 0.5 nmi $< d \leq 5$ nmi : $VAL = 8.4\times d/9 + 4/3$ m
A_{cc}	$d \leq 0.5$ nmi : $A_{cc} = 0.3$ m 0.5 nmi $< d \leq 5$ nmi : $A_{cc} = 2\times d/9 + 17/90$
Satellite constellation	Standard 24 satellite (DO229C) [20]
Location	Honolulu (22°N and 158°W)
Geometry-free multipath time constant	1 minute
Ship multipath time constant	1 minute
Aircraft multipath time constant	30 seconds

Using the described navigation architecture and the parameters given in Table I, Figure 4 shows the vertical positioning estimate error after the initial fix for a single approach using different fixing methods: the conventional method (blue curve), EPIC-only (purple curve), and EPIC+DT (red curve). The black curve in the figure represents the accuracy requirement throughout the approach as defined in Table I. The figure illustrates that, unlike the case-III approach results in [12], the improvement in performance using EPIC compared to the conventional method for this specific Case-I approach is marginal. It is because of the frequent satellite blockages during the approach that the EPIC-only performance is limited and it fails to meet the requirement. However, including the dual track fixing algorithm with EPIC improves the performance substantially and it is evident that there is switching taking place (between 0.5 and 0.7 nmi). It is also evident, that adding the dual track algorithm to EPIC (EPIC+DT) allowed re-fixing the reacquired satellites, resulting in improved accuracy and compliance with the stated requirements. These results are confirmed when all 1440 approaches are considered to compute availability. While the availability using the conventional method is 26.7%, the addition of EPIC boosts the availability to 47.4%. However, when EPIC+DT is used, a remarkable gain in performance is observed with an availability of 97.7%.

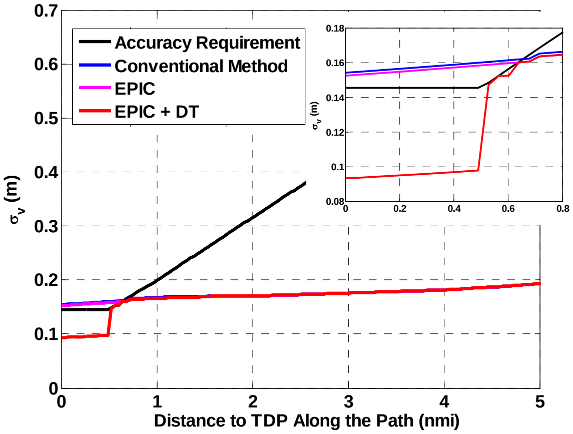

Figure 4: Vertical positioning estimate error using the conventional method, EPIC-only, and EPIC+DT

In order to understand how computationally demanding EPIC is, consider the relative frequency of the number of incorrect fix candidates required to achieve the requirements in the simulations conducted earlier. This analysis is summarized in Figure 5 below. For example, Figure 5 illustrates that in more than 90% of the cases, 20 candidates are sufficient to meet the requirements. However, no more than 500 candidates are needed in any individual case and less than 1% of the cases need more than 100 candidates. Notice that the number of candidates has a direct impact on the memory allocation because the biases are computed using the candidates. For example, the memory should be able to handle 500 bias vectors of at least (3+n) elements (for position and ambiguity only, for example). Remember that all of these biases have to be saved and propagated using Kalman filter measurement and time updates. The simulations conducted in this section were performed using a 3GHz Quad-core Intel Pentium 4, 3GB RAM PC and needed a maximum of 0.07 seconds of EPIC computation time for the 500 candidate case. Therefore, implementing EPIC+DT algorithms in real time applications is quite practical.

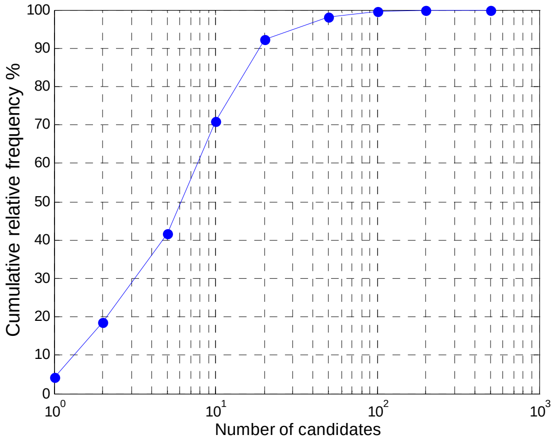

Figure 5 Cumulative relative frequency (the fraction of time the number of incorrect candidates is less than the number on the x-axis) of the sufficient number of candidates required in EPIC

In summary, the conducted simulations show that the EPIC+DT performance is superior to the conventional methods and provides a gain of more than 70% in availability. In the following section, experimental validation of the EPIC+DT algorithm is described.

EXPERIMENTAL VALIDATION

GPS data was collected from two static antennas on the rooftop of the E1-Building at the Illinois Institute of Technology (Chicago, IL). The two antennas were separated by approximately 25 meters. NovAtel OEM-V receivers were used to collect approximately one hour of dual frequency code and carrier phase measurements on December 7, 2009. The data was processed using the shipboard navigation algorithm that was described earlier and explained thoroughly in [19] and [20]. After prefiltering for 10 minutes, the float ambiguity estimates are determined and fixing is attempted using the dual track fixing algorithm. Fixing using EPIC is compared to fixing with the conventional methods. Except for the L2 measurement noise levels, which are inflated because NovAtel OEM-V is not a military receiver, the same parameters in Table I are used in this analysis. Table 2 provides the measurement noise standard deviations that are used in this experimental validation.

Table 2: Measurement noise standard deviations

Parameter	Value
L1 $\sigma_{\Delta\rho}$	0.5 m
L2 $\sigma_{\Delta\rho}$	1.0 m
L1 $\sigma_{\Delta\phi}$	1.0 cm
L2 $\sigma_{\Delta\phi}$	1.8 cm

The true relative position vector between the two antennas has been surveyed using an independent high accuracy algorithm to be 0.448 m, 25.678 m and -0.009 m in local north, east and down directions, respectively. The error in the relative position vector estimates is then computed by subtracting the estimates from the true relative position vector. Errors in the vertical component of the relative position vector estimate using the conventional and EPIC methods are shown in Figure 6 (blue curve). Covariance envelopes using the two methods are also shown in red. It can be seen that the error in EPIC is the same as the conventional method when both algorithms fixed all ambiguities. However, it is clear from the covariance envelopes that EPIC could partially fix more ambiguities 700 seconds earlier (at 600 seconds) than the conventional method (at 1300 seconds). It also provided more accurate estimates with tighter bounds on the position domain errors.

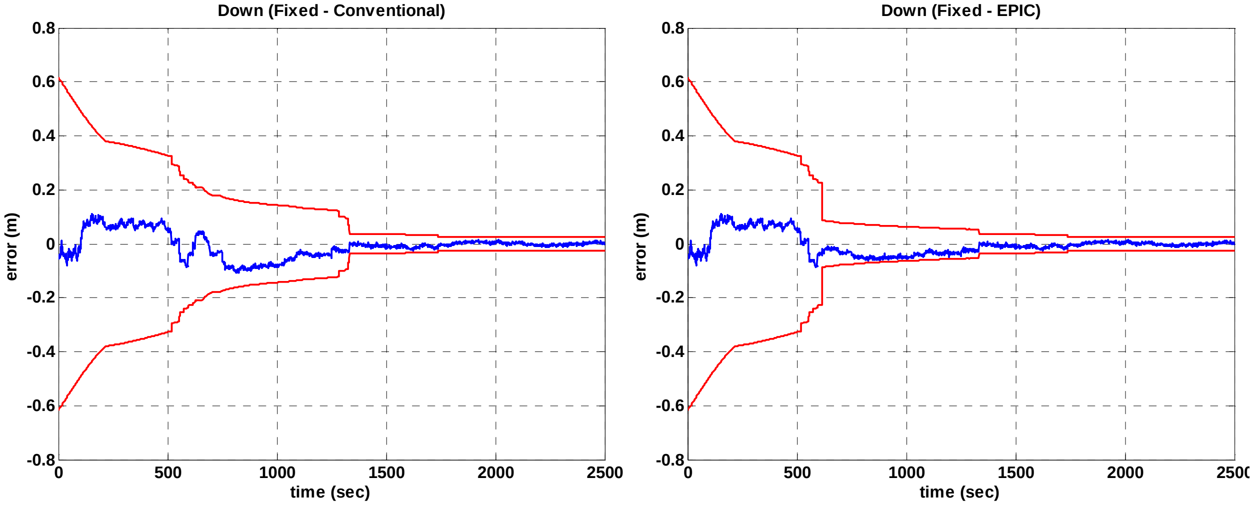

Figure 6: Experimental results using the conventional and EPIC methods

CONCLUSIONS

Cycle ambiguity resolution in applications that require both high accuracy and high integrity is challenging. In previous work, a method capable of meeting stringent integrity and accuracy requirements was developed (EPIC), but it was limited to snap-shot positioning algorithms. In this work, EPIC was expanded to include more general positioning algorithms that include filtering (such as Kalman filtering). In this context, we derived closed form expressions to estimate the time varying position domain cycle resolution integrity risk at every epoch. In addition, a dual track algorithm, which is a mechanism to re-fix new satellites without jeopardizing integrity, has been developed. The advantage of the dual track fixing algorithm is that it does not require any information obtained from prior fixing attempts. Hence, cycle ambiguities are reacquired without accumulating additional integrity risk. The improvement in navigation performance using EPIC with the dual track fixing algorithm was quantified for an example shipboard landing relative navigation application in curved approach scenarios. The availability analysis demonstrated that the new methods provide considerable availability enhancement relative to the best existing method. In addition, computational sensitivity analysis and experimental validation showed that the EPIC and dual track fixing algorithms are both practical for real time applications.

REFERENCES

[1] "Minimum Aviation System Performance Standards for the Local Area Augmentation System Airborne Equipment," RTCA Document Number DO-245A, December 2004.

[2] J. Hansen, G. Romrell, N. Nabaa, R. Andersen, L. Myers, and J. McCormick, "DARPA Autonomous Airborne Refueling Demonstration Program with Initial Results," *Proceedings of the 19th*

International Technical Meeting of the Satellite Division of the Institute of Navigation ION GNSS 2006, Fort Worth, TX, Sep. 2006.

[3] S. Khanafseh and B. Pervan, "Autonomous Airborne Refueling of Unmanned Air Vehicles Using the Global Positioning System," *Journal of Aircraft*, Vol. 44, No. 5, Oct. 2007.

[4] S. Dogra, J. Wright, and J. Hansen, "Sea-Based JPALS Relative Navigation Algorithm Development," *Proceedings of the 18th International Technical Meeting of the Satellite Division of the Institute of Navigation ION GNSS 2005*, Long Beach, CA, Sept. 2005.

[5] M. Heo, B. Pervan, S. Pullen, J. Gautier, P. Enge, and D. Gebre-Eziabher, "Robust Airborne Navigation Algorithm for SRGPS," *Proceedings of IEEE/ION Position, Location, and Navigation Symposium (PLANS '2004)*, Monterey, CA, Apr. 2004.

[6] D. Kim and R. Langley, "GPS Ambiguity Resolution and Validation: Methodologies, Trends and Issues," *Proceedings of the 7th GNSS Workshop - International Symposium on GPS/GNSS,* Seoul, Korea, Nov. 2000.

[7] S. Verhagen, "The GNSS Integer Ambiguities: Estimation and Validation," Ph.D. Dissertation, Delft University of Technology, Delft, Netherlands, Jan. 2005.

[8] P. Teunissen, "GNSS Ambiguity Bootstrapping: Theory and Application," *Proceedings of KIS2001, International Symposium on Kinematic Systems in Geodesy, Geomatics and Navigation,* Banff, Canada.

[9] B. Pervan and F. C. Chan, "System Concepts for Cycle Ambiguity Resolution and Verification for Aircraft Carrier Landings," *Proceedings of the 14th International Technical Meeting of the Satellite Division of the Institute of Navigation ION GPS 2001*, Salt Lake City, UT, Sept. 2001.

[10] M. B. Heo, "Robust Carrier Phase DGPS Navigation for Shipboard Landing of Aircraft," Ph.D. Dissertation, Illinois Institute of Technology, Chicago, IL, Dec. 2004.

[11] S. Khanafseh, B. Kempny, and B. Pervan, "New Applications of Measurement Redundancy in High Performance Relative Navigation Systems for Aviation," Proceedings of the 19th International Technical Meeting of the Satellite Division of the Institute of Navigation ION GNSS 2006, Fort Worth, TX, Sept. 2006.

[12] S. Khanafseh and B. Pervan, "A New Approach for Calculating Position Domain Integrity Risk for Cycle Resolution in Carrier Phase Navigation Systems," *IEEE Transaction on Aerospace and Electronic Systems,* Vol. 46, No. 1, January 2010.

[13] P.J.G Teunissen and S. Verhagen, "The GNSS Ambiguity Ratio-test Revisited: a Better Way of Using it," Survey Review, Vol. 41, No. 312, April 2009.

[14] D. Lawrence, "A New Method for Partial Ambiguity Resolution," Proceedings of the 2009 International Technical Meeting of the Institute of Navigation ION-ITM 2009, Anaheim, CA, Jan. 2009.

[15] P. Teunissen, D. Odijk, and P. Joosten, "A Probabilistic Evaluation of Correct GPS Ambiguity Resolution," *Proceedings of the 11th International Technical Meeting of the Satellite Division of the Institute of Navigation,* Nashville, TN, Sept. 1998.

[16] A. Hassibi and S. Boyd, "Integer Parameter Estimation in Linear Models with Application to GPS," *IEEE Transactions on signal Processing,* Vol. 46, No. 11, Nov. 1998.

[17] P. Misra and P. Enge, Global Positioning System signals, Measurements, and Performance. Lincoln, MA: Ganga-Jamuna Press, 2001.

[18] G. A. McGraw, "Generalized Divergence-Free Carrier Smoothing with Applications to Dual Frequency Differential GPS," *NAVIGATION: Journal of Institute of Navigation*, Vol. 56, No. 2, Summer 2009.

[19] S. Langel, S. Khanafseh, F. C. Chan, B. Pervan, "Cycle Ambiguity Reacquisition in UAV Applications using a Novel GPS/INS Integration Algorithm," *Proceedings of the 2009 International Technical Meeting of the Institute of Navigation ION-ITM 2009,* Anaheim, CA, Jan. 2009.

[20] "Minimum Operational Performance Standards for Global Positioning System/Wide Area Augmentation System Airborne Equipment," RTCA Document Number DO-229C, Nov. 2001, Appendix B.5.

[21] S. Khanafseh, "GPS Navigation Algorithms for Autonomous Airborne Refueling of unmanned Air Vehicles," Ph.D. Dissertation, Illinois Institute of Technology, Chicago, IL, May 2008.