GPS Sensing for Spacecraft Formation Flying [1]

Tobe' Corazzini, Andrew Robertson, John Carl Adams
Arash Hassibi and Jonathan P. How
Stanford University

BIOGRAPHY

Tobe' Corazzini is a graduate student in the Dept. of Aeronautics and Astronautics at Stanford University. She received her BS (1995) from Caltech and MS (1996) from Stanford University.

Andrew Robertson is a Ph.D. candidate in the Dept. of Aeronautics and Astronautics at Stanford University. He received his SB (1993) from MIT and MS (1995) from Stanford University.

John Carl Adams is a Ph.D. candidate in the Dept. of Aeronautics and Astronautics at Stanford University. He received his SB and SM in Aeronautics and Astronautics from MIT.

Arash Hassibi is a Ph.D. candidate in the Department of Electrical Engineering at Stanford University. He received his B.S. (1992) from Tehran Univ. and M.S. (1996) from Stanford University.

Jonathan How is an Assistant Professor in the Dept. of Aeronautics and Astronautics at Stanford University. He received his B.A.Sc (1987) from the University. of Toronto, and SM (1990) and Ph.D. (1993) from MIT, both in the Dept. of Aeronautics and Astronautics.

ABSTRACT

Carrier-phase Differential GPS (CDGPS) for orbit and attitude determination is emerging as a very promising low cost alternative to more conventional methods, such as sun sensors, magnetometers, and star trackers. Relative spacecraft position and attitude determination are important for missions involving formation flying, such as those proposed for stellar interferometry under NASA's New Millennium Program, and LEO missions for coordinated Earth observing. This paper describes research on the development of a GPS based relative position and attitude sensing system in a laboratory environment for future application on spacecraft formations. Previous papers have discussed the demonstration of spacecraft rendezvous and capture in 2D, in a fully functional indoor GPS environment using simulated spacecraft on an air-bearing table. The work presented in this paper extends the previous GPS based sensing to a formation of three prototype spacecraft, and experimentally demonstrates very accurate real-time solutions ($\leq$ 2 cm error) for the three relative vehicle positions. A trade study of estimation algorithms and architectures is presented. Search based techniques for the carrier phase integer ambiguity resolution problem for vehicle formations are also demonstrated on simulation data.

1 INTRODUCTION

Formation flying of multiple spacecraft is an enabling technology for many future space missions [1, 2], and the Global Positioning System (GPS) will play an important role as a sensor. We define formation flying as the coordinated motion control of a group of vehicles where the vehicle positions relative to each other are very important. These vehicles may be groups of trucks, aircraft, or spacecraft. The results in this paper are for prototype spacecraft, but all of these vehicle formation applications share common sensing and control research issues.

Formation flying technologies for spacecraft will enable the use of a 'virtual spacecraft bus' where multiple distributed spacecraft could be coordinated to act as one. This wil enable new scientific missions involving coordinated but distributed measurements, leading to improved stellar interferometry, gravimetry, and synthetic aperture radars. Spacecraft formation control will require a measure of the formation states, *i.e.,* the relative attitude and positions of the vehicles, and GPS offers a promising relative navigation sensor. Our approach is based on the CDGPS techniques that have already been demonstrated on orbit for attitude determination [3]. CDGPS can also provide high precision (cm level) sensing for relative positioning between spacecraft.

Zimmerman has demonstrated the use of CDGPS for spacecraft rendezvous and capture between two simulated spacecraft in an indoor laboratory GPS environment [4]. To date, much of the experimentation in formation flying on this testbed has been focused on the selection of controller architectures [5], including centralized and distributed control schemes. The work in this paper addresses the dual problem of investigating an estimator architecture.

The problem of estimating the formation state lends itself

[1] Presented at Institute of Navigation GPS-97, Kansas City, Missouri, September 1997

to the same architecture choices available in the control problem: decentralized or centralized. Centralized estimation involves one vehicle collecting the GPS measurements from all other vehicles, and solving for the state of the entire formation. This provides *global* knowledge of the entire formation, but with a large number of vehicles, the computational demands on one vehicle may make this approach impractical. In addition, a large formation requires high communication rates transferring large amounts of raw data, which could lead to bottlenecks in some centralized estimation architectures.

A distributed scheme involves individual vehicles estimating their own states relative to other vehicles in the formation using only a subset of the total number of measurements available. The key benefit of the decentralized estimation is that it avoids the excessive processor and communications demands associated with the centralized architecture. Decentralized estimation distributes the computational load among all of the vehicles, thus yielding an implementable solution for very large formations. A further benefit of the decentralized approach is that it can be expanded easily when a new vehicle joins the formation or if one is lost because of a mechanical failure. Of course, the key issue to be examined is whether the distributed approach provides sufficient sensing accuracy for formation control. For example, if one vehicle can measure its absolute position (using GPS or other sensors), and all other vehicles can only measure their position relative to a neighbor, what level of estimation accuracy results for the formation? This research uses experimental and simulated results to demonstrate the capability to maintain accurate knowledge of the formation state as a whole, while using a decentralized sensing architecture.

2 FORMATION FLYING TESTBED

To study formation flying issues related to the precision formation flying mission proposed by JPL [2], a formation flying testbed has been created in the Stanford Aerospace Robotics Laboratory. It consists of 3 active free-flying vehicles that move on a 12 ft $\times$ 9ft granite table top (see Figure 1). These air cushion vehicles simulate the zero-g dynamics of a spacecraft formation in a plane. The vehicles are propelled by compressed air thrusters. Each vehicle has onboard computing and batteries, and communicates with the other vehicles via a wireless ethernet, making them self-contained and autonomous.

An indoor GPS environment was constructed for the experiments by Zimmerman [4]. This consists of six ceiling mounted pseudolite transmitters broadcasting the L_1 carrier phase signal modulated by a unique C/A code, with no navigation data. One transmitter is designated the *master pseudolite* and transmits a 50 bps data message modulated on top of the C/A code, which contains timing information for synchronizing the carrier phase measurements of multiple receivers in the laboratory to within

Fig. 1: Formation Flying Testbed

one millisecond epoch. It is this synchronization that enables differential carrier phase measurements to be made between receivers on separate vehicles. Each vehicle has a six channel Trimble TANS Quadrex receiver with customized software. There is an overhead vision system mounted above the table, and each object on the table is tagged with a unique LED pattern that can be tracked. The vision system has an absolute accuracy better than 1 cm for position and 1.5° for attitude throughout the workspace.

Various estimation architectures are analyzed in this paper using both static and dynamic tests. Static tests are used to validate the architecture alone, with no additional movement of the system. In the dynamic tests, two of the vehicles are commanded to control position relative to their neighbor, while one maintains its absolute position with an observed positioning error of approximately 10 cm in both X and Y. This test is critical for the indoor, near-transmitter environment because as the vehicles undergo even small motions,the signal to noise ratios received from each of the transmitters change greatly. Occlusion of the signal to one vehicle by another is also a common occurrence. These are some of the expected challenges of a self-constellation arrangement in deep space, and must be examined. The following section outlines the estimation problems to be solved. It is followed by experimental results on the formation flying testbed.

3 ESTIMATION PROBLEM

The measured carrier phase at antenna j of vehicle i from pseudolite k is:

$$\phi_{ijk} = |D_{ijk}| + c\tau_{vi} + c\tau_{pk} + \lambda K_{ijk} \qquad (1)$$

where the vector D_{ijk} is simply a vector from the phase center of the pseudolite antenna to the phase center on each receive antenna. The terms $c\tau_{vi}$ and $c\tau_{pi}$ represent the portion of the phase incurred by clock errors between the transmitter and the receiver, and are the dominating terms in the measurement equations. Differencing over multiple measurements is used to eliminate these terms.

The intra-vehicle single differences contribute primarily to determination of the attitude of each vehicle. These

measurements are obtained by differencing between the master antenna ($j = m$) and each of the slave antennas j of vehicle i for measurement from pseudolite k:

$$\Delta\phi_{ijk} = |D_{imk}| - |D_{ijk}| + \lambda M_{ijk} \tag{2}$$

$\forall k$ and $\forall j \neq m$. In this expression, the M_{ijk} are the intra-vehicle integers.

The inter-vehicle double differences contribute primarily to the determination of the relative positions between each vehicle. Given N pseudolites, there are $N-1$ unique double differences between pseudolites k_1 and k_2 ($k_1 \neq k_2$). These differences are calculated in order to eliminate the remaining effects due to clock differences $c(\tau_{v1} - \tau_{v2})$

$$\begin{aligned}\nabla\Delta\phi_{ijk_1k_2} &= |D_{imk_1}| - |D_{jmk_1}| \\ &-(|D_{imk_2}| - |D_{jmk_2}|) + \lambda N_{ijk_1k_2}\end{aligned} \tag{3}$$

$\forall k_1$, k_2 with $k_1 \neq k_2$. In this expression, $N_{ijk_1k_2}$ refers to the inter-vehicle integers.

The double difference measurements are coupled to the states of the two vehicles used in each pairing, so all of the measurements must be combined to resolve the entire formation state. From equations (2), (3), and the quaternion constraints ($\epsilon_{i1}^2 + \epsilon_{i2}^2 + \epsilon_{i3}^2 + \epsilon_{i4}^2 = 1$) , the complete set of measurements can be related to the vehicle states:

$$\begin{bmatrix} \Delta\phi_{1jk} \\ 0 \\ \Delta\phi_{2jk} \\ 0 \\ \Delta\phi_{3jk} \\ 0 \\ \nabla\Delta\phi_{12k_1k_2} \\ \nabla\Delta\phi_{23k_1k_2} \end{bmatrix} = \begin{bmatrix} h_1(X_1) \\ h_c(X_1) \\ h_2(X_2) \\ h_c(X_2) \\ h_3(X_3) \\ h_c(X_3) \\ h_{12}(X_1, X_2) \\ h_{23}(X_2, X_3) \end{bmatrix} + \lambda \begin{bmatrix} M_{1jk} \\ 0 \\ M_{2jk} \\ 0 \\ M_{3jk} \\ 0 \\ N_{12k_1k_2} \\ N_{23k_1k_2} \end{bmatrix} \tag{4}$$

where h_i is a set of nonlinear functions of X_i, h_{12} is a set of nonlinear functions of the given formation states, h_c is a quaternion constraint function. Given the phase measurements and a solution of the integer ambiguities, the optimal estimate of the formation states $X_i(t)$ can be solved in real-time using an extended Kalman filter. The algorithms and typical results for a formation of three vehicles are presented in the following sections.

4 EXPERIMENTAL RESULTS

For the experiments presented in this paper, the computer onboard each vehicle executes an algorithm to solve for its position and attitude in an absolute frame, using knowledge of the location of six overhead pseudolites. The overhead vision system provides a measure of truth, indicating absolute position of the vehicles on the table. Estimation of the vehicle state is done in a circular chain: vehicle 1 solves for the positions of itself and vehicle 2, vehicle 2 solves for itself and vehicle 3, and vehicle 3 solves for itself and vehicle 1. While these vehicles solve for the absolute positions of the vehicles in the pair, what is more important for the formation sensing and control are the relative positions which can be found by simply subtracting the two absolute position estimates. In general it is expected that this sensing system will provide a more accurate estimate of the relative position between vehicles because many of the most important error sources are predominantly common-mode and their effects are removed by differencing the absolute position estimates.

Since the most significant metric in this work is the relative accuracy between various vehicles, it is desirable to assess the cumulative positioning error in the formation. One measure of cumulative error can be seen by fixing vehicle 1 at a known position, and using its calculation of relative position to locate vehicle 2. Then, starting at this new location, the relative position calculation of vehicle 2 is used to locate vehicle 3. Finally, using this estimate of the third vehicle's location together with the relative position calculation of vehicle 3, a second position estimate is found for vehicle 1. We can now compare the known starting position of vehicle 1 to the position estimated at the end of this chain and use this as a measure of the cumulative error. Figure 2 shows this error for the static case using 35 sec of measured data. The figure compares the mean GPS solution against the vision position solution. Note that while the first calculation (relative location of vehicle 2 to vehicle 1) is extremely accurate, the error grows as the calculations are continued around the formation. The worst case mean error for the data as plotted in Figure 2 is 3.0 cm (radial distance), with a variance of only 0.4 cm^2. For this same data set, Figure 3 shows the two-norm of all errors in relative position and attitude, for all three vehicles. The mean of this metric is 2.93 cm.

A second performance metric includes the relative errors in position for all three vehicles. In this case we calculate:

$$J = \frac{1}{3}\sum_{i=1}^{3}\sqrt{\Delta X_i^2 + \Delta Y_i^2} \tag{5}$$

For the case in Figure 2, the metric J yields a mean formation error of only 1.62 cm per vehicle.

Closer examination of the sensing between vehicles shows that the mean errors in relative position are typically less than 2 cm in X and Y. Figure 4 shows a typical example of the error in the relative position between vehicles 2 and 3 when compared to the truth sensor. Some data points are seen to lie outside of the 2 cm bounds. However, the mean error is -1.7 cm in X and -1.4 cm in Y, with a variance of 0.13 cm^2 in X and 0.16 cm^2 in Y. The actual distribution of the errors is shown in Figures 5 and 6. These histograms of the relative error show that 97% of the measurements have less than a 2 cm error in Y and

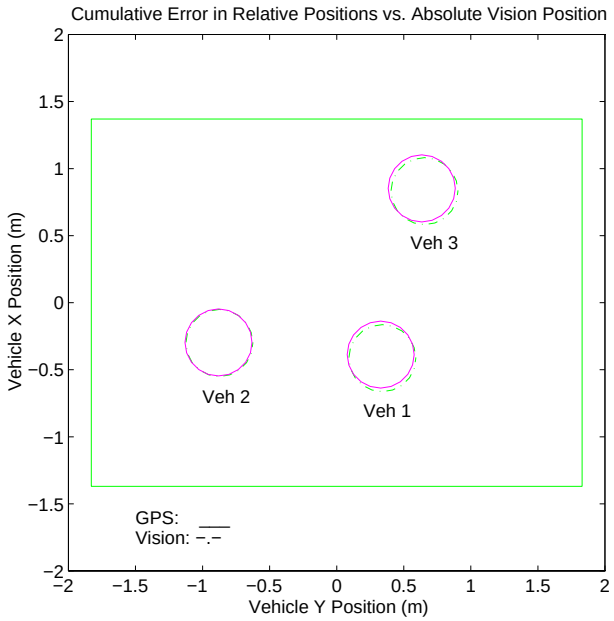

Fig. 2: Formation error compared to vision, starting with vehicle 1, and proceeding around the formation using relative position calculations.

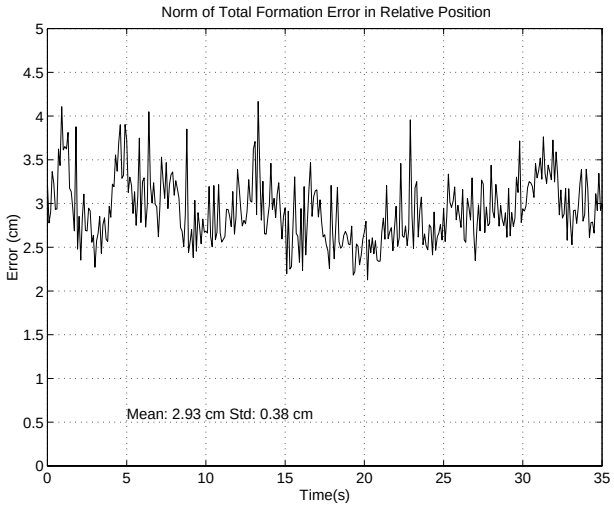

Fig. 3: Total error in the formation $\|R\|_2$.

86% have less than a 2 cm error in X.

Similar tests were also performed with the vehicles moving. In this case, one vehicle was commanded to move around a small area of the table while the other two executed loose formation keeping control algorithms to track this vehicle. Figure 7 compares the accuracy of the GPS solution to the vision truth while the vehicles move. Although the signal to noise ratios change as the vehicles move, the plot shows that the mean of the error in relative position is only 0.91 cm, with a variance of 0.76 cm^2. Figures 8 and 9 show the distributions of the errors in X and Y. The Y axis error is seen to be more evenly distributed than in the static case, but for both X and Y, Figures 8 and 9 show that the majority of the errors are small.

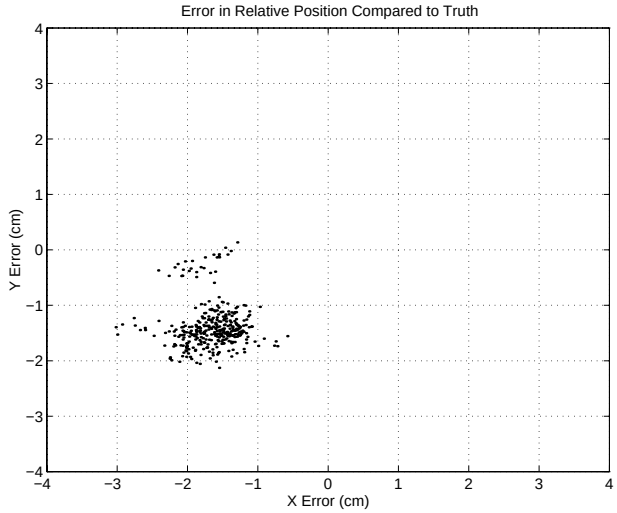

Fig. 4: Error in the difference of the GPS and vision relative position solutions (X and Y) for vehicles 2 and 3.

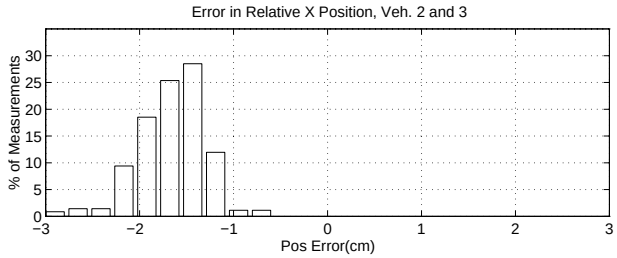

Fig. 5: Histogram of GPS relative position errors in X compared to vision.

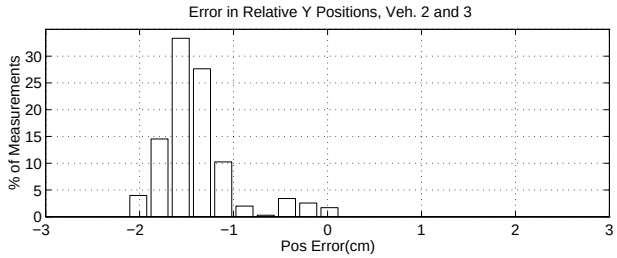

Fig. 6: Histogram of GPS relative position errors in Y compared to vision.

These dynamic tests are important because as the vehicles move around, the signal to noise ratios change dramatically, and the vehicles are prone to dropping and gaining pseudolite signals. In some tests, this changing of configurations (from one pseudolite set used in calculations to another) occurred as often as every two seconds. However, by using the predicted values of the positions to reinitialize the integers after losing a pseudolite, the vehicles can continue with the same estimation algorithms. Experimental results using 4, 5, and 6 pseudolites were all found to be consistent in the mean value of errors in relative position. The most serious degradation in errors results from using signals with very low signal to noise ratio (SNR). Using the same average error metric J for a case of all three vehicles moving under low SNR conditions, the average error increases to 2.9 cm.

These results also demonstrate that in both the dynamic

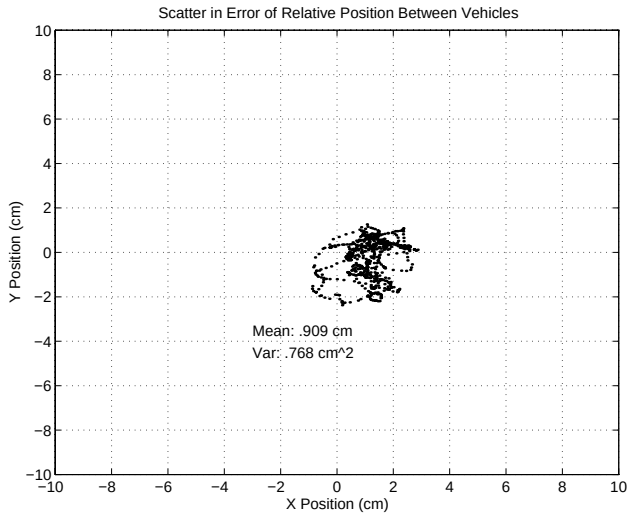

Fig. 7: Error in difference of relative position solutions from GPS and vision, vehicle in motion

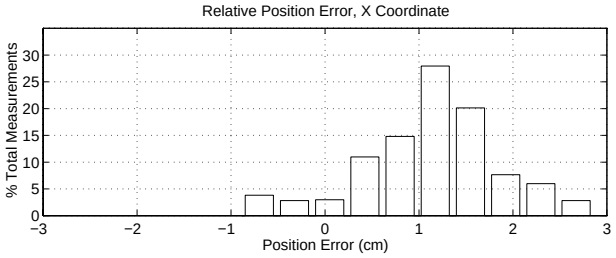

Fig. 8: Histogram of error in relative position in X

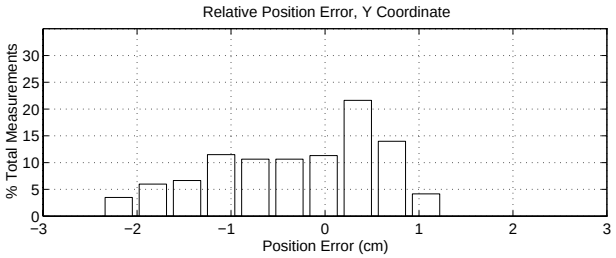

Fig. 9: Histogram of error in relative position in Y

and static testing, the attitude calculations are consistent with the vision system measurements. The mean of the error in the GPS relative attitude as measured by the vision system is 0.25°. Since the vision system accuracy is on the order of 1.0°, the GPS attitude solution error is indistinguishable from vision system error.

The particular formation flying application will dictate the acceptable tolerances on relative position errors. But with per-vehicle formation errors on the order of 2 cm, these experiments show the validity of decentralized estimation. These results also show that if one vehicle can estimate its absolute position, and each subsequent vehicle solves for the relative position to a neighbor, the error introduced in the formation would be acceptable for many missions.

Note that these experimental results are limited by multipath. Thus, these errors are a worst-case expectation for deep space, where the multipath environment is improved. Careful calibration of the antenna baselines will also reduce the errors. Also, the main errors in the position estimates are biases, which are in part attributable to the vision system. Thus, a calibration of the vision system must be done to get a better idea of how much of the bias is due to the GPS sensor.

5 ESTIMATION TRADE STUDY

Given the choice of CDGPS as the sensor for a formation of space vehicles, an estimation architecture to process the GPS data must be chosen. The previous section presented the experimental implementation of a decentralized architecture. This section considers other architectures and estimation algorithms which could be used for this system. The objective of this trade study is to investigate the relative performance, computational effort, and communication requirements of three estimation algorithms and of three estimation architectures.

A computer simulation is the basis of the algorithm and architecture comparisons. The simulation code runs the estimation calculations as they would be performed onboard the vehicles. The estimators update the vehicle positions and attitudes with new GPS measurement data every 0.1 sec for 10 sec. Both simulated and measured GPS phase data are used in this analysis.

5.1 Estimator Requirements

The formation flying testbed simulates a formation of three space vehicles with a nearby constellation of six GPS pseudolites. The estimation process uses the available GPS measurements to solve for the position and attitude estimates for each vehicle. The accuracy of these estimates is very important for precise formation control. Relative accuracy is particularly important in the position solution. Due to the error sources in a GPS sensing system, relative position solutions are often more accurate than the separate vehicle absolute position solutions, as shown in [4]. To enhance the robustness of the sensing system, it is also important that the formation estimation architecture should continue to function after one vehicle in the formation fails. The resources (computational and communication) required by the estimation architecture determine its cost to the spacecraft design.

The formation flying testbed presents several challenges beyond those found in the standard GPS estimation problem [4, 6]. The tasks of initializing and tracking the carrier wave integers is assumed to be handled separately from the estimation process (see Section 6). The spherical wavefront geometry is modeled and included as a nonlinear function in the estimation loop. These measurement nonlinearities are expected to play an important role for formations with nearby GPS transmitters [2].

5.2 Estimation Algorithms

Standard estimation algorithms exist for solving the problem of estimating a state based on measurements plus sys-

tem and measurement models [7]. Three algorithms were considered: 1) an extended Kalman filter, 2) an iterated weighted least squares fit, and 3) an iterated extended Kalman filter.

These algorithms are representative of different estimation choices. Weighted least squares does not rely on a model of the system dynamics, but simply finds the state that yields the best fit for each new set of data. The extended Kalman filter takes into account a dynamic model and tracks confidence from measurement to measurement. Iterated versions of these converge faster initially to a solution but are more costly computationally. The iterated weighted least squares algorithm was used in the experiment discussed in Section 4 of this paper.

Note that for our 3 vehicle system, the estimation problem has 21 states to estimate (3 positions and 4 attitude quaternions per vehicle) and 69 measurements (GPS carrier phase single and double differences). So, for example, the linearized version of the measurement matrix H is 69 $\times$ 21 with elements that are non-linear functions of the vehicle positions. The iterated estimation algorithms recalculate H during each cycle. The raw data arrives at 10 Hz, so these estimation computations must be performed well within 0.1 second.

The simulation code was used to compare the three algorithms for the case of a two vehicle formation. In the simulation, initial position and attitude estimates were picked and the diagonal elements of the initial covariance matrix P_0 were set to large values to indicate low confidence in the initial conditions. The measurement noise variance values R were either based on measured SNR values (for the runs with real GPS measurement data) or selected based on typical noises measured on the formation flying testbed (double difference noise variances $R_{dd} = 4 \times 10^{-5}\text{m}^2$ and single difference noise variances $R_{sd} = 4 \times 10^{-6}\text{m}^2$). Relatively large amounts of process noise, Q, were included to avoid filter divergence (variance of 0.001 for each state). For the iterated algorithms, a maximum number of iterations (20) and a convergence criterion (norm change in estimate < 0.001) were specified (same for both iterated algorithms).

5.3 Estimation Architectures

The vehicles in this testbed cannot individually calculate their own position and attitude using only their own GPS phase measurements. However, groups of two and three vehicles can form position and attitude estimates from their combined GPS measurements. These are called *pairs* and *trios*, and are the base units for the estimation architectures considered. Note that for a three vehicle formation, one trio estimator is the centralized estimation strategy, but for a formation of 30 vehicles, many trio estimators could be used, and this would be a decentralized strategy.

We considered three estimation architectures: (a) two pairs with one vehicle doing all calculations, (b) three pairs with each vehicle doing one pair-estimation (used for the experiment in Section 4), and (c) one trio. These architectures have differing computational and communication requirements. For instance, architecture (b) requires all vehicles have computers to do estimation, whereas cases (a) and (c) require only one vehicle be capable of estimation. This represents the extent of the architectures possible on the current testbed, but clearly these basic configurations would play an important role in the larger formations that have been proposed [8].

5.4 Comparison

The above algorithms and architectures were compared using a MATLAB simulation script. The simulation performs the estimation steps while tracking the number of floating point operations performed (using the *flops* command). The accuracy of the solution is measured by the mean and standard deviation of the errors in the estimates of the vehicle positions and attitudes. Relative position and attitude errors are also recorded.

Verifying the GPS Measurement Model. The input to the simulation code is the GPS carrier phase single-difference and double-difference measurements from the vehicles. This data was simulated for the two- and three-vehicle cases. The simulated data contains zero-mean white noise. The variances of the noise added to the single- and double-differences were given values typical of those observed on the formation flying testbed. The noise on the actual measurement signals is dominated by multipath which is difficult to model accurately and is not white. Thus the simulation was also run on actual measurements to verify that similar accuracies are attained. As shown in Table 1, the cases run with real data showed performance that is quite similar to the simulated cases. The simulation runs each algorithm on ten seconds worth of (10 Hz) GPS phase measurement data collected on the formation flying testbed.

Comparing Estimation Algorithms. The estimation algorithms discussed above were implemented for the two vehicle (one pair) case. Note the relative position accuracies in Table 1 compared to the absolute position accuracies. As discussed above, GPS-based sensors are particularly well suited for relative position sensing. Looking particularly at the extended Kalman filters in the active vehicle case, the relative position estimate is more than seven times better than the absolute position estimate. The attitude estimates do not show the same level of improvement because they are primarily based on the single difference measurements which do not contain the same degree of common-mode cancellation as do the double differences. Also note that while the vehicles are moving, the Kalman filter absolute position estimate errors average 2.4 cm with a 0.35 cm standard deviation, which corresponds

Table 1: Analysis of average measurement error for three estimator algorithms. The real data cases use GPS single and double difference measurements collected on the formation flying testbed.

CASE	Alg.	Mean MFLOPS per step	Absolute				Relative			
			Position (cm)		Angle (deg)		Position (cm)		Angle (deg)	
			mean	σ	mean	σ	mean	σ	mean	σ
Static Simulated	EKF	0.43	1.50	0.85	0.16	0.45	0.63	0.56	0.18	0.19
	IWLS	1.51	1.70	0.84	0.12	0.06	0.61	0.41	0.16	0.11
	IEKF	1.01	1.40	0.72	0.11	0.06	0.59	0.40	0.16	0.11
Static Real Data	EKF	0.54	1.00	0.23	0.06	0.03	0.27	0.15	0.10	0.08
	IWLS	1.17	2.20	1.60	0.13	0.09	0.99	0.82	0.18	0.17
	IEKF	1.00	1.00	0.23	0.06	0.03	0.27	0.15	0.10	0.08
Dynamic Real Data	EKF	0.54	2.40	0.35	0.33	0.22	0.32	0.18	0.50	0.39
	IWLS	1.18	2.70	2.00	0.20	0.14	0.91	0.83	0.31	0.27
	IEKF	1.11	2.40	0.35	0.33	0.22	0.32	0.18	0.50	0.39

Table 2: Average measurement errors. Mean MFLOPS per step show requirements for each vehicle performing estimation.

Arch.	Mean MFlops per step	Absolute				Relative			
		Pos. cm		Ang. deg		Pos. cm		Ang. deg	
		mean	σ	mean	σ	mean	σ	mean	σ
2 Pairs	2.1	1.40	0.52	0.23	0.09	0.84	0.46	0.38	0.18
3 Pairs	1.0	1.50	0.49	0.35	0.12	1.20	0.60	0.51	0.21
1 Trio	2.8	1.10	0.48	0.22	0.09	0.60	0.24	0.37	0.18

to quite good performance. These numbers also agree very well with the experimentally observed performance in Section 4. The attitude estimation is also very good with the mean angle error never more than half a degree.

Comparing Estimation Architectures. The three three-vehicle estimation architectures discussed above are compared using the simulation code. These initial comparisons are made for a station-keeping example wherein all three vehicles are not moving. The results are shown in Table 2 for the three architectures on 10 sec of measurements.

Note the one trio architecture requires the most computational resources, but is also the most accurate estimator. The table also shows that the three pairs architecture requires each vehicle to do estimation calculations and seems to be the least accurate. The two pairs and one trio architectures are centralized, whereas the three pairs is decentralized.

The communication requirement may be viewed several ways. In general, decentralized architectures require more GPS measurement data flowing between the robots. Whereas the central vehicle in a centralized architecture must send vehicle state estimates once calculated back out to the formation. Communication bottlenecks occur when a large amount of information must reach a vehicle before the estimation process can begin. This is the metric for our trade study. Centralized architectures are more susceptable to bottlenecks than are decentralized architectures.

The centralized architectures run the risk of a single-point formation failure if a problem occurs on the central spacecraft. The central spacecraft will be the most complicated because of the computational demand placed on it, so will be a likely place for failure. Decentralized architectures do not run this risk. Therefore, decentralized architectures are thought to be more robust, and should be far more amenable to adding new vehicles.

5.5 Conclusions

Tuning a particular estimator algorithm/architecture could improve the performance (i.e. R, Q, and iteration loop parameters could be adjusted). For this paper, the estimator parameters were all fixed at reasonable values so there would be a fixed basis for the comparisons given below.

Estimation Algorithms. The IWLS is the most computationally expensive estimator because it typically requires more iterations to converge than does the IEKF. Of course, the (non-iterated) EKF is the cheapest filter. These results show that the EKF performance is comparable to the IEKF (even in the case where the vehicles are moving). The IEKF and IWLS converge faster than the EKF from a poor initial guess. Once the estimates converge to close to truth, the EKF performs as well as the IEKF for less computational cost.

Estimation Architectures. The results show that more total computation is needed for the three pairs case, but this is spread equally among the robots. More architectures are possible with more than three vehicles, but the results presented can be extrapolated to these cases using pairs and trios as the building blocks. For instance, we could maximize robustness and accuracy, but increase the risk of a communication bottleneck by having each robot do a trio estimation calculation.

Trade Study. The extended Kalman filters clearly outperform the weighted least squares algorithm in computational cost and accuracy; they are better suited to the estimation of a dynamic system state. The extended Kalman filter performs well, but for this system the iterated version provides more accuracy and can converge faster with a poor initial estimate. Therefore, the iterated extended

Table 3: Comparison of estimator architectures.

Arch.	Case	Accuracy	Comp[1]	Comm[2]	Robustness
Centralized	Pair	●●●	●●●	●●●●	●
	Trio	●●●●●	●●●●●	●●●●	●
Decentralized	Pair	●●	●	●●	●●●●●
	Trio[3]	●●●●●	●●●●●	●●●●	●●●●●

1) Computation requirement for vehicles with estimators.
2) Communication bottlenecks.
3) Case not simulated. Results inferred from the other analysis.

Kalman filter is preferred for this application.

Table 3 presents a general summary of the architecture trade study. As is often the case, there is no clear winner among the candidates. Other factors, particularly the use of the estimated states in motion planning and control may dictate a preferred architecture. The communication requirement may need to be examined more closely for the many-vehicle formation. A communication protocol must be established to allow smooth information flow among the vehicles to avoid bottlenecks. The robustness of the decentralized approach to vehicle loss is very useful. This study supports the experimental results in the previous section in that it confirms that the decentralized estimation architecture is a viable alternative for this small formation.

6 INITIALIZATION

As discussed in Section 3, an important part of the GPS formation sensing problem is the initialization of the carrier phase measurement integer ambiguities in Eq. 4. The integer resolution problem has been studied extensively for the GPS attitude determination problem, with the work falling into two categories of solution methods: *instantaneous search* methods and *motion based* methods. Search methods use a snapshot of carrier phase data and search over the carrier phase integers to minimize some cost function. Motion based methods collect carrier phase data over time as the vehicle moves, and use the change in measurement geometry between the antenna baselines and transmitter line of sight to form an over-determined batch estimation problem.

The spacecraft formation sensing problem has some important differences from the attitude estimation problem on a single vehicle in terms of the initialization of the carrier phase integer ambiguities. First, since the phase measurements are from separate receivers on each vehicle, a time bias must be accounted for between the receivers and the pseudolite transmitters. In our experiment, double differenced phase measurements are used to remove the time biases from the receivers and transmitters, so the corresponding phase integers are actually double differenced phase integers. This is important to keep in mind, because any loss of lock and subsequent re-acquisition of a transmitter by a given receiver requires that all double difference measurements using that phase measurement have their integers re-initialized.

A second difference between the attitude and formation problems is that formations do not have some of the simplifying constraints of the attitude problem. For example, no assumptions can be made about the general direction that the antenna array is pointing. Furthermore, in this case we cannot assume that the baselines between all antennas are constant. These factors complicate the integer search techniques because these assumptions are typically used to reduce the attitude integer search space. Also, it is not as easy to put a vehicle formation through the large motions that change the geometry significantly with respect to the transmitters as it is for the attitude motion of a single vehicle.

In our previous work on the formation integer initialization problem [6], we demonstrated the use of a motion based nonlinear batch least squares estimation technique for the resolution of the double differenced carrier phase integers with both simulated and experimental data. Since our testbed has a near constellation of pseudolite transmitters, the necessary geometry change for batch integer resolution was feasible, but required motions that could be impractical in terms of time and propellant costs.

With this motivation, we investigated this issue further to study the performance of two integer search methods for the formation phase integer ambiguity resolution problem. In particular, we applied the algorithm designed by Knight [9] for solving the attitude integer problem on the Trimble TANS Vector attitude receiver. We also applied the global integer search method described by Hassibi [10] to the formation problem. The simulation data presented is for both an instantaneous integer search with the vehicles held static, and for a dynamic case where the search was applied to a batch of double difference phase data.

The batch analysis was developed as an attempt to combine the strengths of integer search techniques and motion based techniques. Search techniques can provide solutions quickly without requiring vehicle formation maneuvers, but are prone to false solutions when the measurements are noisy. Motion based techniques are more robust to false solutions, but require more time to compute.

6.1 Knight Algorithm

The integer resolution algorithm defined by Knight searches the possible integer combinations for the best fit of state estimate and integer estimate for minimizing the quadratic cost function in Eq. 6, where x is the formation state, N is the double difference integers, H is the linearized measurement matrix, and R is the measurement variance. The minimization is split into two problems; minimizing the cost over the state estimate and over the integer estimate (see Eq. 7). For the attitude integer problem, the state estimation is equivalent to the measurement

update in a Kalman filter, while for the formation integer problem, it is equivalent to the *extended* Kalman filter measurement update. The Knight algorithm makes use of the fact that the same cost calculated by the Kalman filter during state estimation can be used recursively, with no additional computation required, in the integer estimation.

$$W = \sum_{i=1}^{m} \frac{(\nabla\Delta\phi_i(k) - H_i x_{est} + \lambda N_{est})}{R_i} \quad (6)$$

$$\min_{\underline{N}} [\min_{\underline{x}} W(\underline{x}, \underline{N})] \quad (7)$$

Simply searching the set of all possible integers and looking for the one with the lowest cost is infeasible due to the enormous number of possible integer combinations. To reduce the integer search tree, the Knight algorithm eliminates any branch of the integer search tree that increases the cumulative quadratic cost. This approach is valid and works well for the linear problem of attitude integer estimation. For our formation integer problem, the nonlinear measurement model invalidates this assumption. All integer search techniques are prone to errors given noise in the measurement, but even when the integer search was limited to a narrow range around an *a priori* state estimate, false integer solutions were found that gave the lowest quadratic cost.

6.2 Hassibi Algorithm

The integer search method developed by Hassibi formulates the problem as a maximum likelihood estimation problem for the vehicle state estimate and measurement integers, and a verification problem for determining the probability of a correct integer solution. The vehicle states are real valued and the phase integers are integer valued. This combined maximum likelihood estimate problem in Eq. 8 can be recast as the *integer* least squares estimation problem in Eq. 9. The integer estimation problem can be interpreted geometrically as a lattice search where the goal is to find the closest integer lattice point for a given state estimate. The verification problem is then one of determining the probability that the vehicle is within a certain decision boundary around that lattice point.

$$\left(\hat{\underline{x}}, \hat{\underline{N}}\right) = \min_{(\underline{x},\underline{N}) \in R^p \times Z^m} \|\underline{y} - h(\underline{x}) - \lambda \underline{N}\| \quad (8)$$

$$\Rightarrow \quad \hat{\underline{N}} = \min_{\underline{N} \in Z^m} \|\tilde{\underline{y}} - G\underline{N}\| \quad (9)$$

6.3 Simulation Results

Two sets of simulations were run to examine the performance of the integer estimation algorithms. The first set of simulations were for the static case, where the three modelled vehicles in our testbed were stationary on the granite table. In this case a set of simulations was executed for random initial conditions on the initial state estimate error covariance. The initial error covariance was varied, and the subsequent average norm of the integer estimate error was found by averaging over 20 runs.

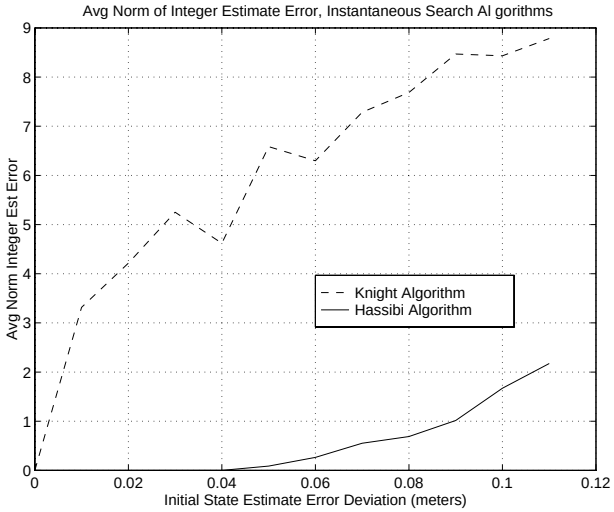

Fig. 10: Average norm integer estimate error vs. error in initial state, static case.

The second set of simulations was for the dynamic case, where the three vehicles are moving across the table in formation at 20 cm/sec. The differential phase data was collected at 1 Hz sampling rate, and used with the Hassibi algorithm in a batch solution.

Static Case. The results of the static case are presented in Figure 10. The plot shows the integer estimate error for the two static algorithms, and shows that the Knight algorithm has a nonzero average integer estimate error even for small errors in the initial state estimate. The Hassibi algorithm successfully estimates the integers for initial state estimate error deviations less than 4 cm, but has a nonzero above that.

The CPU time required for integer resolution is greater for the Knight algorithm, and it increases with the initial condition error, where the Hassibi algorithm has a relatively constant cpu time burden for increasing initial condition error.

Two conclusions can be drawn from these static tests. First, the instantaneous search using a single set of double difference phase measurements will only work when the error in the initial condition for the state estimate error is small (< 4 cm). Since our testbed has an absolute positioning accuracy with a deviation of 10 cm, this performance is probably not adequate. The second conclusion is care must be taken when applying integer resolution techniques that were designed for the attitude integer problem to the formation integer problem.

Dynamic Case. The region of zero integer estimate error shown for the static Hassibi algorithm in the static case can be extended by taking data over time and solv-

ing for the integers in a batch version of the same algorithm. The results of the dynamic simulations are presented in Figures 11 and 12. The zero integer estimate error range was extended to the necessary 10 *cm* of error on the initial state estimate by increasing the number of batch measurements to 30. The error curves show that the integer estimate error can be reduced by applying a search technique to a batch set of measurements collected over time from a moving vehicle formation.

The convergence properties of the batch Hassibi algorithm are plotted in Figure 12. In this result, a new measurement was collected each second, and a new batch solution for the integer estimates was calculated using the Hassibi search algorithm. Note that as the number of measurements increases, the error in the state estimate and integer estimate do not decrease uniformly, and actually increase at some points when measurements are added. As an output of the lattice search, an upper and lower bound on the probability of correct integer estimation is also calculated. The second plot in figure 12 shows the upper bound on this probability converging to 1. This bound gives a metric for deciding when the integer least squares process needs to be run, and also when enough data has been collected for the batch solution.

7 CONCLUSIONS

In conclusion, this paper has shown that it is feasible to extend the CDGPS techniques to provide very accurate (approximately 2 cm) relative navigation sensor for a formation of three vehicles. The numerical analysis presented indicates that several improvements could be made to the experimental setup, including converting to iterated extended Kalman filters and including the dynamic Hassibi algorithm for integer initialization. These results provide increased confidence in future applications of GPS for formation flying spacecraft.

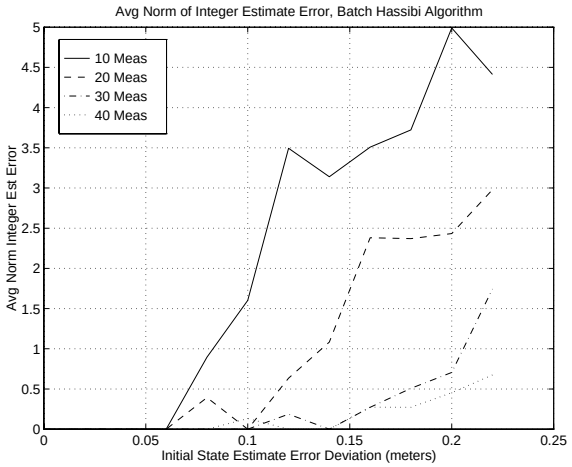

Fig. 11: Average norm integer estimate error vs. error in initial state, dynamic case.

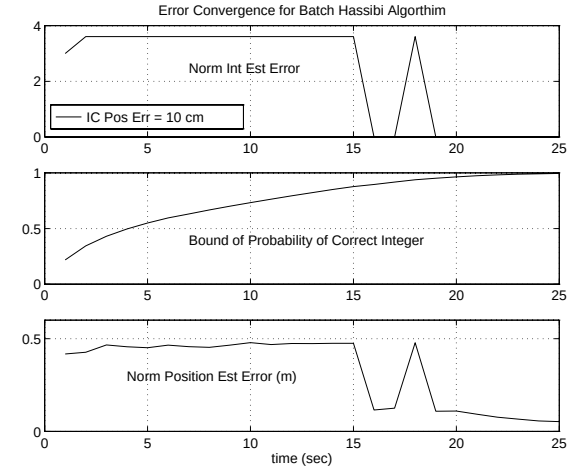

Fig. 12: Error convergence of integer estimate, batch Hassibi algorithm.

References

[1] F. Bauer, J. Bristow, D. Folta, K. Hartman, D. Quinn, and J. P. How, "Satellite Formation Flying Using an Innovative Autonomous Control System (AUTOCON) Environment," in *Proceedings of the AIAA GNC Conf*, (New Orleans, LA), May 1997.

[2] K. Lau, S. Lichten, and L. Young, "An Innovative Deep Space Application of GPS Technology for Formation Flying Spacecraft," in *Proceedings of the AIAA GNC Conf*, (San Diego, CA), July 1996.

[3] E. G. Lightsey, *Development and Flight Demonstration of a GPS Receiver for Space*. Dept. of Aeronautics and Astronautics, Stanford University, Stanford, CA, 94305, Jan. 1997.

[4] K. R. Zimmerman and R. H. Cannon Jr., "Experimental Demonstration of GPS for Rendezvous Between Two Prototype Space Vehicles," in *Proceedings of the ION GPS-95 Conf*, (Palm Springs, CA), Sept 1995.

[5] A. Robertson, T. Corazzini, E. LeMaster, and J. P. How, "Formation Sensing and Control Technologies for a Separated Spacecraft Interferometer," Submitted to the *ACC*, (San Diego, CA), March 1998.

[6] J. C. Adams, A. Robertson, K. Zimmerman, and J. How, "Technologies for Spacecraft Formation Flying," in *Proceedings of the ION GPS-96 Conf*, (Kansas City, MO), Sept. 1996.

[7] A. Gelb, *Applied Optimal Estimation*. Cambridge, MA: MIT Press, 1974.

[8] M. Colavita, C. Chu, E. Mettler, M. Milman, D. Royer, S. Shakian, and J. West, "Multiple Spacecraft Interferometer Constellation (MUSIC)," Tech. Rep. JPL D-13369, JPL - Advanced Concepts Program, February 1996.

[9] D. T. Knight, "A New Method for Instantaneous Ambiguity Resolution," in *Proceedings of the Institute of Navigation 7th Int'l Tech. Mtg Conf*, (Salt Lake City, Utah), Sept. 1994.

[10] A. Hassibi and S. Boyd, "Integer Parameter Estimation in Linear Models with Applications to GPS," in *Proceedings of 35th IEEE CDC*, (Kobe, Japan), Dec. 1996.