Accuracy/Computation Performance of a New Trilateration Scheme for GPS-Style Localization[1]

Kar-Ming Cheung
Jet Propulsion Laboratory
4800 Oak Grove Dr.
Pasadena, CA 91109
818-393-0662
Kar-Ming.Cheung@jpl.nasa.gov

Glenn Lightsey
Georgia Institute of Technology
North Ave NW
Atlanta, GA 30332
404-385-4146
glenn.lightsey@gatech.edu

Charles Lee
Jet Propulsion Laboratory
4800 Oak Grove Dr.
Pasadena, CA 91109
818-354-8197
Charles.H.Lee@jpl.nasa.gov

Abstract—We recently introduced a new geometric trilateration (GT) method for GPS-style positioning. Preliminary single-point analysis using simplistic error assumptions indicates that the new scheme delivers almost indistinguishable localization accuracy as the traditional Newton-Raphson (NR) approach. Also, the same computation procedure can be used to perform high-accuracy relative positioning between a reference vehicle and an arbitrary number of target vehicles. This scheme has the potential to enable a) new mission concepts in collaborative science, b) in-situ navigation services for human Mars missions, and c) lower cost and faster acquisition of GPS signals for consumer-grade GPS products.

The new GT scheme differs from the NR scheme in the following ways:

1. The new scheme is derived from Pythagoras Theorem, whereas the NR method is based on the principle of linear regression.
2. The NR method uses the absolute locations (x_i, y_i, z_i)'s of the GPS satellites as input to each step of the localization computation. The GT method uses the Directional Cosines U_i's from Earth's center to the GPS satellite S_i.
3. Both the NR method and the GT method iterate to converge to a localized solution. In each iteration step, multiple matrix operations are performed. The NR method constructs a different matrix in each iterative step, thus requires performing a new set of matrix operations in each step. The GT scheme uses the same matrix in each iteration, thus requiring computing the matrix operations only once for all subsequent iterations.

In this paper, we perform an in-depth comparison between the GT scheme and the NR method in terms of a) GPS localization accuracy in the GPS operation environment, b) its sensitivity with respect to systematic errors and random errors, and c) computation load required to converge to a localization solution.

TABLE OF CONTENTS

1. INTRODUCTION

As of June 30, 2017, the United States' Global Positioning System (GPS) infrastructure consisted of 31 operational satellites [1]. These satellites provide 24/7 global location and timing services for users on Earth's surface and in low Earth orbit (LEO). The cost of development and deployment of GPS is estimated to be about $33 billion, and the annual operation and maintenance cost is about $1 billion [2]. Yet the economic benefits of GPS are tremendous; it is estimated that the monetary benefits of GPS to the US economy in 2013 alone is about $56 billion [3].

In addition to economic benefits, GPS is changing the everyday life of people in the areas of technology, culture, and thinking. There is no end in sight as to how GPS can be integrated with other technologies, and its infusion revolutionizes and enables many commercial, space, and military applications and services.

GPS provides 3-dimensional (3-D) position estimates via trilateration, which refers to the general technique of computing position based on measurement of distances. The standard GPS trilateration scheme is expressed in terms of distance measurements and positions in an Earth-centered Cartesian coordinate system. The set of simultaneous equations is of the form

$$d_i = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} + c\Delta t \quad i = 1, \cdots, n \quad (1)$$

where (x, y, z) is the position of vehicle V to be estimated, (x_i, y_i, z_i) are known positions of the GPS satellites S_i [2], and n is the number of satellites. Δt is the clock bias between V and the GPS time standard, which is maintained by the GPS operation segment. c is the speed of light. In the GPS trilateration computation, (x, y, z) and $c\Delta t$ can be solved uniquely for $n \geq 4$. The standard approach to solve the system of equations in (1) is known as the Newton-Raphson method, which is a general iterative method that uses linear regression to find the root of a function [2].

We recently introduced a new geometric trilateration (GT) method for GPS-style positioning [4]. Preliminary single-point analysis using simplistic error assumptions indicates that the new scheme delivers almost indistinguishable localization accuracy as the traditional Newton-Raphson

(NR) approach. Also, the same computation procedure can be used to perform high-accuracy relative positioning between a reference vehicle and an arbitrary number of target vehicles [5]. This scheme has the potential to enable a) new mission concepts in collaborative science, b) in-situ navigation services for human Mars missions, and c) lower cost and faster acquisition of GPS signals for consumer-grade GPS products.

The new GT scheme differs from the NR scheme in the following ways:

1. The new scheme is derived from Pythagoras Theorem, whereas the NR method is based on the principle of linear regression.

2. The NR method uses the absolute locations $(x_i, y_i, z_i)'s$ of the GPS satellites as input to each step of the localization computation. The GT method uses the Directional Cosines $U_i's$ from Earth's center to the GPS satellites S_i.

3. Both the NR method and the GT method iterate to converge to a localized solution. In each iteration step, multiple matrix operations are performed. The NR method constructs a different matrix in each iterative step, thus requires performing a new set of matrix operations in each step. The GT scheme uses the same matrix in each iteration, and thus requires computing the matrix operations only once for all subsequent iterations.

The rest of the paper is organized as follows: Section 2 reviews the NR method and the GT scheme for trilateration. Section 3 introduces a notional navigation satellite system architecture for the Human Mars Exploration Missions. The detailed system concept is described in [6]. Using the Human Mars landing site scenario, we compare the accuracy performances between the NR and the GT schemes in Section 4, and the computation performances are compared in Section 5. Section 6 provides concluding remarks and discusses future work.

2. REVIEW OF NEWTON-RAPHSON (NR) SCHEME AND GEOMETRIC TRILATERATION (GT) SCHEME

2.1 REVIEW OF NEWTON-RAPHSON SCHEME

Newton-Raphson's iterative method and its convergence are based on the approach of linear regression. Let $\vec{P} = (\tilde{x}, \tilde{y}, \tilde{z})$ be the estimated location for a given iteration. A residual location $\Delta\vec{P} = (\Delta x, \Delta y, \Delta z),$ and an estimated clock offset $\Delta = c\Delta t$ are computed by solving the following equation:

$$\Delta\vec{P}' = (G^TG)^{-1}G^T \vec{d}$$

where $\Delta\vec{P}' = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Lambda \end{bmatrix}$, and $\vec{d} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix}_{n \times 1}$ for $n \geq 4$.

The matrix G is of the form

$$G = \begin{bmatrix} \dfrac{x_1 - \tilde{x}}{d_1} & \dfrac{y_1 - \tilde{y}}{d_1} & \dfrac{z_1 - \tilde{z}}{d_1} & -1 \\ \dfrac{x_2 - \tilde{x}}{d_2} & \dfrac{y_2 - \tilde{y}}{d_2} & \dfrac{z_2 - \tilde{z}}{d_2} & -1 \\ \vdots & \vdots & \vdots & \vdots \\ \dfrac{x_n - \tilde{x}}{d_n} & \dfrac{y_n - \tilde{y}}{d_n} & \dfrac{z_n - \tilde{z}}{d_n} & -1 \end{bmatrix}_{n \times 4}$$

The estimated location $(\tilde{x}, \tilde{y}, \tilde{z})$ is then updated as $(\tilde{x}, \tilde{y}, \tilde{z})$ + $(\Delta x, \Delta y, \Delta z) \rightarrow (\tilde{x}, \tilde{y}, \tilde{z})$. Note that:

1. The matrix G is constructed using the GPS satellite locations (x_i, y_i, z_i) as well as the estimated location $(\tilde{x}, \tilde{y}, \tilde{z})$ of P for a given iteration.

2. The first three entries of row i in G correspond to the unit vector from the intermediate location $(\tilde{x}, \tilde{y}, \tilde{z})$ of each iterative step to the GPS satellite S_i.

3. The estimated location $(\tilde{x}, \tilde{y}, \tilde{z})$ is different in each iterative step, thus the matrix G is different, and the complicated computation of $(G^TG)^{-1}G^T$ has to be performed in each step.

The details of this method can be found in many GPS books, e.g. [2].

2.2 REVIEW OF GEOMETRIC TRILATERATION SCHEME

The Geometric Trilateration method iterates and provides coverage to a localization solution based on alternating applications of Pythagoras Theorem in its iteration process. We formulated the problem as follows: Let E denote the center of the planetary body with coordinate $(0, 0, 0)$. Consider three points V, E, and S_1 that form a triangle Δ_1 in the Euclidean space as shown in Figure 1. Let r_1 be range between E and S_1, and r_1' be the pseudo-range measurements between V and S_1. We consider the presence of the clock bias Δt between the vehicle V and the GPS satellites $S_i's$, $1 \leq i \leq n$. We assume that the clocks of the GPS satellites are perfectly synchronized. We express the unknown clock bias of the vehicle V with respect to S_1 as an unknown correction factor $\Lambda = c\Delta t$ in the pseudo-range measurements r_1'. The same correction factor Δ occurs in all other pseudo-range measurements r_i', $1 \leq i \leq n$.

2

The detailed derivation is derived in [5], and the computation procedures are summarized in Figure 1.

r_i' includes the delay Δ for $i = 1,...,n$ $n \geq 4$

then $\vec{U}_i \circ \dot{P} = d_i = r_i - r_i'' = r_i - \sqrt{(r_i'-\Delta)^2 - s_i^2}$

or $\vec{U}_i \circ \vec{P} - \Delta = r_i - \sqrt{(r_i'-\Delta)^2 - s_i^2} - \Delta$

or $A\dot{\vec{P}} = \vec{d}$;

where $A = \begin{bmatrix} \vec{U}_1^T & -1 \\ \vdots \\ \vec{U}_n^T & -1 \end{bmatrix}_{n\times 4}$; $\dot{\vec{P}} = \begin{bmatrix} x \\ y \\ z \\ \Delta \end{bmatrix}_{4\times 1} = \begin{bmatrix} \vec{P} \\ \Delta \end{bmatrix}_{4\times 1}$;

$\vec{d} = \begin{bmatrix} r_1 - \sqrt{(r_1'-\Delta)^2 - s_1^2} - \Delta \\ \vdots \\ r_n - \sqrt{(r_n'-\Delta)^2 - s_n^2} - \Delta \end{bmatrix}_{n\times 1}$

$\dot{P} = (A^T A)^{-1} A^T \vec{d}$;

$\Lambda_1 = S_1 VE$

Scheme:

$s_i^2 = \begin{cases} 0 & \text{1st iteration} \\ \|\vec{P}\|^2 - (\vec{U}_i \circ \vec{P})^2 & \text{afterward} \end{cases}$ $\Delta = \begin{cases} 0 & \text{1st iteration} \\ P'(4) & \text{afterward} \end{cases}$

$d_i = r_i - \sqrt{(r_i'-\Delta)^2 - s_i^2} - \Delta$

$\vec{P}' = (A^T A)^{-1} A^T \vec{d}$
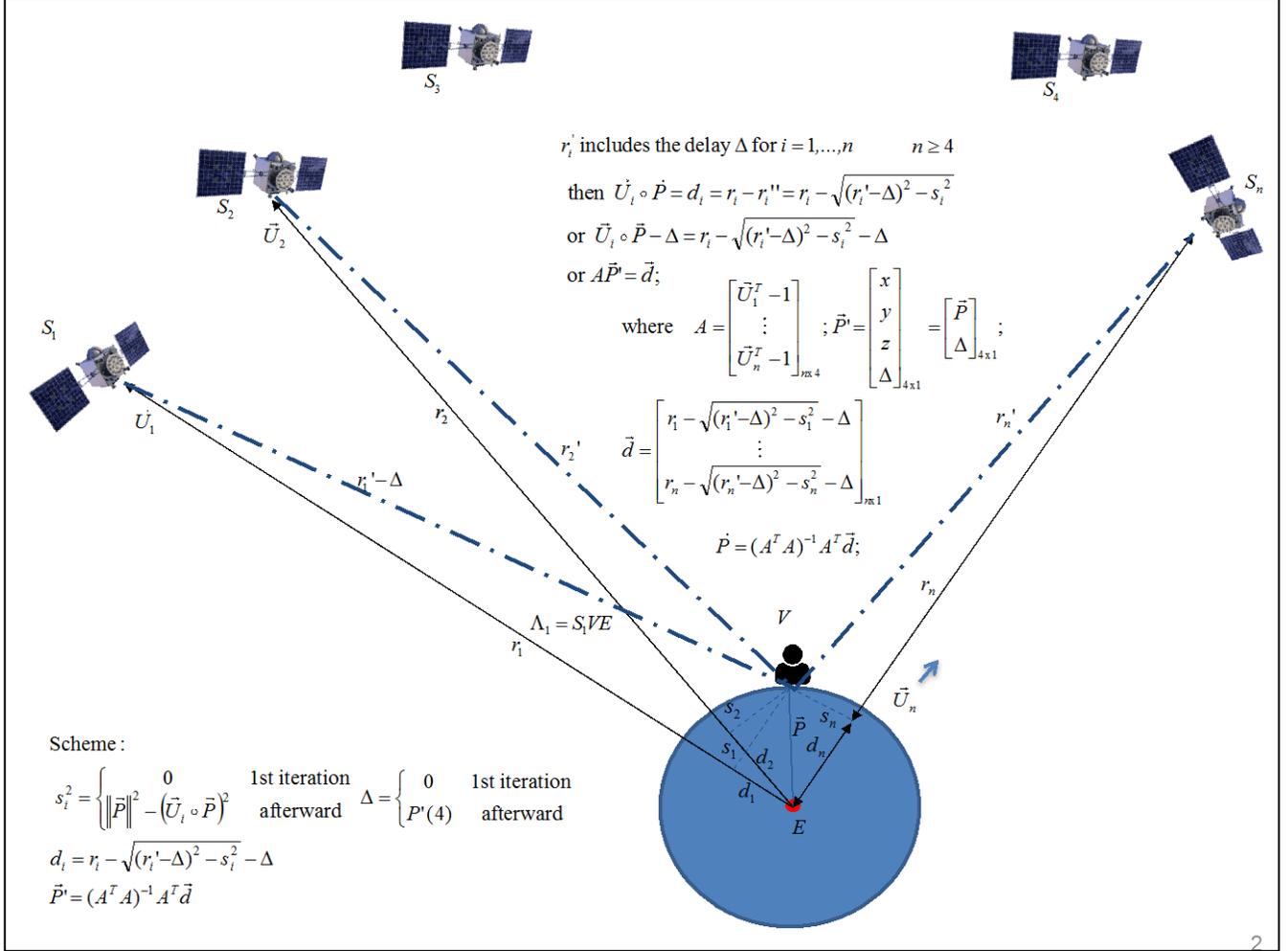
Figure 1. Iterative Procedure of the GT Scheme

3   A NOTIONAL MARS NAVIGATION SATELLITE SYSTEM ARCHITECTURE

We consider the scenario of the Human Mars landing site at Utopia Planitia on Mars, and propose a navigation satellite constellation that provides navigation and timing services in the surrounding region of the landing site. The navigation satellite constellation leverages on the two planned areostationary relay orbiters and the Deep Space Habitat in a circular 48-hour inclined orbit, and augmented it with a navigation satellite in an areosynchronous orbit that traces around a figure-8 path. The Mars orbiters' orbital parameters are given as follows:

− Aerostationary orbiter 1 (Areo45): 162.5° due East
− Aerostationary orbiter 2 (Areo90): 207.5° due East
− Aerosynchronous orbiter (Areo68):180° due East, 20° inclined
− Deep Space Habitat (Mars48hr): 180° due East, 149.5° inclined

The orbits of the Mars navigation nodes are shown in Figure 2 (3-D view), and the projections of these orbits onto the Mars surface are shown in Figure 3 (2-D view). Note that in Figure 3, the Mars navigation nodes cluster together, and Utopia Planitia is north of the cluster. The satellite-receiver geometry appears to be weak and the geometric dilution of precision (GDOP) is high. In other words, the localization solution can be very sensitive to the errors in the raw-range measurements. We describe a system concept that uses the same trilateration scheme to perform both absolute positioning and relative positioning in [6], and we show the simulation results and detailed error analysis in [7].
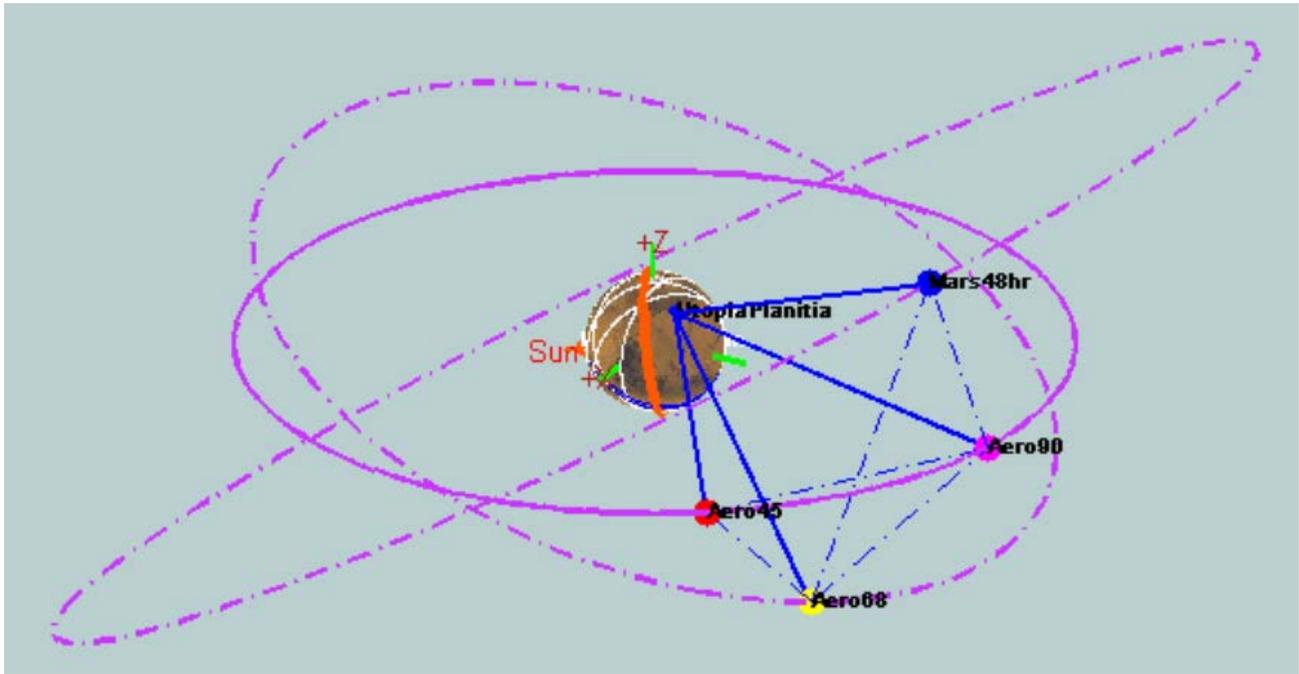
**Figure 2. Orbits of the Notional Mars Navigation Nodes (3-D View)**
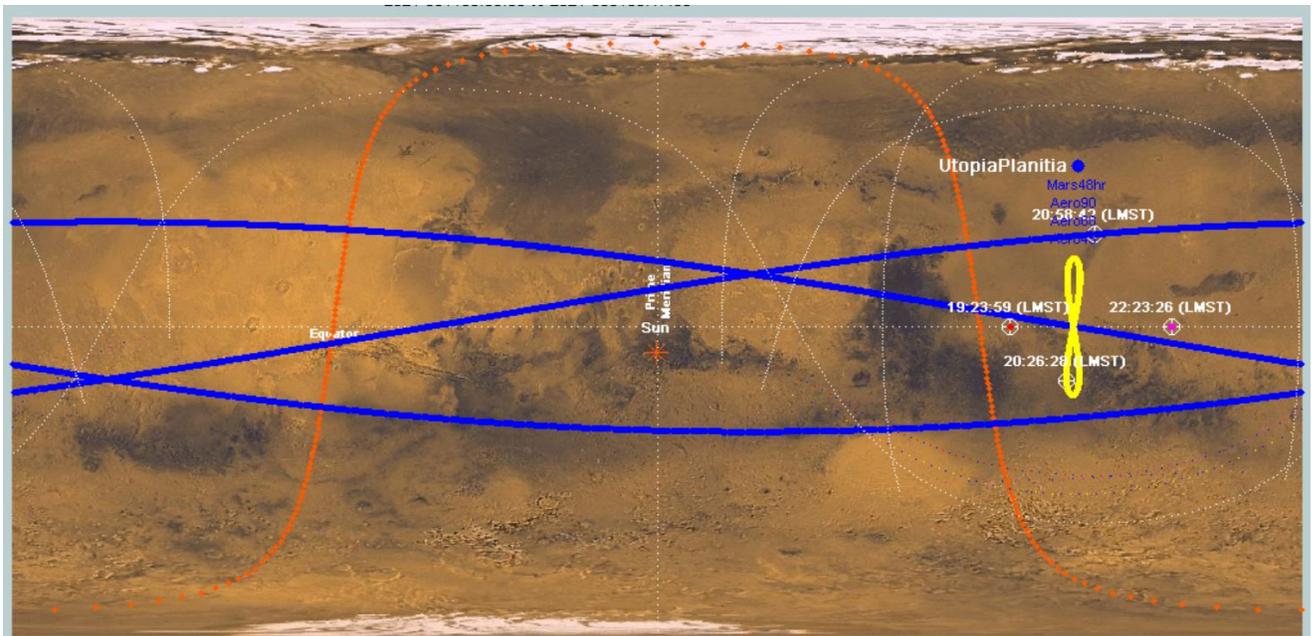
Figure 3. Orbits of the Notional Mars Navigation Nodes (2-D View)

4.   ACCURACY PERFORMANCE COMPARISON

The traditional Newton-Raphson and new Geometric Trilateration methods were compared using the Martian navigation scenario that was presented in the previous section.

In this analysis, both navigation node errors and receiver range estimation errors are considered. These errors serve as proxies to model the most common error types in modern satellite navigation systems. For example, ~~random~~ navigation node errors model imperfect knowledge in the transmitting satellite locations and clock offsets. Receiver range estimation errors model uncorrected environmental effects such as transmission medium delays, multipath, and receiver noise.

Each navigation node has a true distance d_i which is known within an error given by $d_i' = d_i + v_i$ where each v_i is an independent normally distributed random variable with mean

4

μ and standard deviation σ_v, i.e., $v_i \sim N(\mu, \sigma_v^2)$. In actuality v_i is the norm of a random vector perturbation in the coordinates of transmission node (x_i, y_i, z_i):

$$d_i' = \sqrt{(x_i + v_{xi})^2 + (y_i + v_{yi})^2 + (z_i + v_{zi})^2} \qquad (2)$$
$$i = 1, \cdots, n$$

With

$$v_i = \sqrt{v_{xi}^2 + v_{yi}^2 + v_{zi}^2} \qquad i = 1, \cdots, n \qquad (3)$$

And each $v_{xi}, v_{yi}, v_{zi} \sim N(0, \sigma_v^2/3)$.

Each receiver pseudo-range measurement is assumed to have a statistically independent random measurement error due to receiver noise, with a normal standard deviation σ_r that is simulated at a specified value. In other words, each pseudo-range estimation is given by:

$$r_i' = d_i' + \varepsilon_i \qquad i = 1, \cdots, n \qquad (4)$$

With $\varepsilon_i \sim N(0, \sigma_r^2)$.

A position solution was obtained using both algorithms for 10,000 simulations of the Martian position described in the previous section with the statistical receiver noise errors for pseudo-range and each navigation node location. A range of different error conditions is shown for pseudo-range measurement error σ_r from 0 to 5.0 cm and navigation node position error σ_v from 0 m to 35 m. A receiver clock offset of $\varDelta t = 10$ microseconds was included in every simulation.

The results for the traditional NR algorithm are shown in Table 1. It is instructive to consider the traditional NR algorithm localization error performance for several of the cases that are listed in Table 1. In the upper left corner, it is seen that when both the navigation node error and the pseudo-range error are zero, the algorithm determines the correct position as expected. In the leftmost column of the Table 1, when the pseudo-range position error is given by $\sigma_r = 1.0$ cm,

the standard deviation of the 3D localization error is σ_{3D} =112.74 cm. This value is very close to the theoretically predicted localization accuracy by the position dilution of precision (PDOP) geometry figure of merit which is given by the square root of the trace of the first three elements of the geometry matrix product inverse, $(G^T G)^{-1}$. For the geometry given in this problem, PDOP = 113.17, which predicts a 3D localization error of σ_{3D} =113.17 cm in the simulated case given by $\sigma_r = 1.0$ cm. This result agrees with the simulated result to within 0.4%. Finally it is noted that for navigation node position errors as small as $\sigma_v = 0.5$ m, this effect dominates the overall error statistics such that all the entries in each column of Table 1 are approximately the same regardless of the value of pseudo-range measurement error. This result suggests that for the range of values studied, it is more beneficial to minimize the navigation node position error to improve the overall localization performance. In the case given with navigation node error $\sigma_v = 1$ m and pseudo-range error $\sigma_r = 5.0$ cm, the 3D position localization error standard deviation is $\sigma_{3D} = 65.64$ m. For the range of error values considered in this study, σ_{3D} performance scales approximately linearly with navigation node error standard deviation σ_v.

The geometric trilateration (GT) algorithm described in Section II was compared to the traditional NR algorithm under identical error conditions. The results of the 3D position localization error for the GT algorithm are shown in Table 2. For each case simulated with the NR algorithm in Table 1, an identical simulation was performed using the same statistical sequences as measurement inputs to the GT algorithm. In other words, the ensemble statistics of each cell in Table 2 are identical to those used in the corresponding cell in Table 1. This allows a fair comparison of the two algorithms and removes any effects that may be attributed to variations in the statistics between the two cases.

Traditional NR Algorithm		Navigation Node Error, σ_v							
		0 m	0.5 m	1 m	2 m	5 m	10 m	30 m	35 m
	0 cm	0.00	3273.85	6547.69	13095.39	32738.48	65476.99	196431.3	229169.9
	0.10 cm	11.27	3273.70	6547.54	13095.23	32738.32	65476.82	196431.1	229169.7
	0.25 cm	28.19	3273.56	6547.35	13095.01	32738.08	65476.58	196430.9	229169.5
	0.50 cm	56.37	3273.51	6547.12	13094.69	32737.71	65476.19	196430.5	229169.1
	1.00 cm	112.74	3274.15	6547.03	13094.24	32737.04	65475.45	196429.7	229168.3
	2.00 cm	225.48	3278.35	6548.30	13094.06	32735.98	65474.10	196428.1	229166.7
	5.00 cm	563.71	3313.95	6563.76	13099.34	32735.15	65471.23	196423.9	229162.4

Table 1. σ_{3D} Localization Error standard deviation (cm) of the Traditional NR Scheme. PDOP=113.17.

Alternate GT Algorithm	Navigation Node Error, σ_v							
	0 m	0.5 m	1 m	2 m	5 m	10 m	30 m	35 m
0 cm	0.00	3273.85	6547.69	13095.39	32738.48	65476.99	196431.3	229169.9
0.10 cm	11.27	3273.70	6547.54	13095.23	32738.32	65476.82	196431.1	229169.7
0.25 cm	28.19	3273.56	6547.35	13095.01	32738.08	65476.58	196430.9	229169.5
0.50 cm	56.37	3273.51	6547.12	13094.69	32737.71	65476.19	196430.5	229169.1
1.00 cm	112.74	3274.15	6547.03	13094.24	32737.04	65475.45	196429.7	229168.3
2.00 cm	225.48	3278.35	6548.30	13094.06	32735.98	65474.10	196428.1	229166.7
5.00 cm	563.71	3313.95	6563.76	13099.34	32735.15	65471.23	196423.9	229162.4

Table 2. σ_{3D} Localization Error standard deviation (cm) of the New GT Scheme. PDOP=113.17.

As is seen by comparing Tables 1 and 2, the navigation performance of the alternate GT algorithm is *exactly the same* as that which was obtained using the traditional NR algorithm. When both algorithms are presented with the same navigation node errors and receiver noise errors, they converge to the same position solution. This conclusion is intuitively satisfying–both algorithms are working with exactly the same information, so they ought to achieve the same numerical solution. The results of Tables 1 and 2 validate the GT algorithm against the NR algorithm as an alternate method for obtaining a navigation solution based on trilateration.

5. COMPUTATION PERFORMANCE COMPARISON

Although both the GT and NR algorithms achieve identical navigation solutions when provided with the same information as seen in the previous section, the computational method by which they obtain these results is different. Therefore, the computational performance of the two algorithms was compared. As was stated in Section 1, both methods perform matrix inversions and both use iterate procedures to converge to a solution. However, the NR method computes its matrix inversion inside its iterative loop, requiring an inversion operation to be performed at every iteration.

The root mean squared (RMS) execution time of each position solution was recorded for each of the test cases over the 10,000 navigation simulations listed in the previous section. Tables 3 and 4 show that the NR algorithm consistently converged within to its solution within 0.01 cm repeatability in 6 iterations and approximately 330 microseconds on a standard laptop computer. The computational cost of the NR algorithm on the test computer is about 55 microseconds per iteration. The execution times were obtained using the *tic* and *toc* functions in Matlab on a laptop computer. These values are machine-dependent and are intended for comparison purposes only between the methods.

Traditional NR Algorithm	Navigation Node Error, σ_v							
	0 m	0.5 m	1 m	2 m	5 m	10 m	30 m	35 m
0 cm	6	6	6	6	6	6	6	6
0.10 cm	6	6	6	6	6	6	6	6
0.25 cm	6	6	6	6	6	6	6	6
0.50 cm	6	6	6	6	6	6	6	6
1.00 cm	6	6	6	6	6	6	6	6
2.00 cm	6	6	6	6	6	6	6	6
5.00 cm	6	6	6	6	6	6	6	6

Table 3. RMS Iteration Count of the NR Algorithm.

Traditional NR Algorithm	Navigation Node Error, σ_v							
	0 m	0.5 m	1 m	2 m	5 m	10 m	30 m	35 m
0 cm	330.18	336.21	329.02	331.08	331.37	331.25	330.67	331.43
0.10 cm	331.16	329.47	331.22	333.19	331.12	330.13	330.95	334.07
0.25 cm	330.38	332.01	338.44	331.06	331.34	330.84	331.60	332.16
0.50 cm	330.48	331.30	334.67	332.06	332.54	332.67	334.23	330.85
1.00 cm	329.04	331.25	331.45	330.46	332.39	332.33	330.82	332.53
2.00 cm	331.49	330.84	332.67	334.54	335.20	330.39	337.95	331.21
5.00 cm	337.32	329.75	331.37	331.63	331.33	330.23	333.06	336.86

Table 4. RMS Execution Time (microsec) of the NR Algorithm.

By comparison, the GT method performs its inversion of the matrix $(A^T A)^{-1}$ only once, because the product is a function of the initial conditions of the guess only and does not have to be updated inside the iterative loop. However, the total GT computation time is longer, owing to the slower convergence rate of the algorithm.

Tables 5 and 6 show that the NR algorithm consistently converged within to its solution within 0.01 cm repeatability in 36 iterations and approximately 1260 microseconds. The computational cost of the GT algorithm is about 35 microseconds per iteration.

Alternate GT Algorithm	Navigation Node Error, σ_v							
	0 m	0.5 m	1 m	2 m	5 m	10 m	30 m	35 m
0 cm	36	36	36	36	36	36	36	36
0.10 cm	36	36	36	36	36	36	36	36
0.25 cm	36	36	36	36	36	36	36	36
0.50 cm	36	36	36	36	36	36	36	36
1.00 cm	36	36	36	36	36	36	36	36
2.00 cm	36	36	36	36	36	36	36	36
5.00 cm	36	36	36	36	36	36	36	36

Table 5. RMS Iteration Count of the GT Algorithm.

Alternate GT Algorithm	Navigation Node Error, σ_v							
	0 m	0.5 m	1 m	2 m	5 m	10 m	30 m	35 m
0 cm	1260.52	1299.69	1268.90	1271.10	1269.04	1268.33	1269.61	1264.37
0.10 cm	1272.40	1298.85	1262.08	1281.08	1271.93	1269.91	1264.77	1266.85
0.25 cm	1270.05	1268.28	1261.68	1277.05	1270.06	1273.51	1262.43	1278.33
0.50 cm	1265.67	1266.82	1270.54	1273.19	1266.38	1270.74	1272.08	1276.25
1.00 cm	1271.30	1261.87	1262.61	1273.36	1270.47	1269.68	1284.95	1270.61
2.00 cm	1285.71	1262.21	1273.69	1263.89	1266.73	1270.55	1267.84	1275.50
5.00 cm	1288.90	1266.10	1266.49	1266.28	1276.33	1265.72	1269.76	1271.35

Table 6. RMS Execution Time (microsec) of the GT Algorithm.

The convergence rates of the NR and GT methods are compared for noise-free measurements in Figure 4. Each algorithm's convergence is shown with the solution residual on the vertical axis and the iterate number on the x-axis. Starting with the same initial position residual, the NR algorithm is seen to converge at a faster than an exponential rate, such that it has met the cutoff convergence criteria within only 4 iterations. Whereas the GT convergence rate is exponential, and requires 14 iterations to achieve the same cutoff convergence criteria. This difference in the rates of convergence of the two algorithms explains the longer total execution time of the GT algorithm when compared to the NR algorithm.
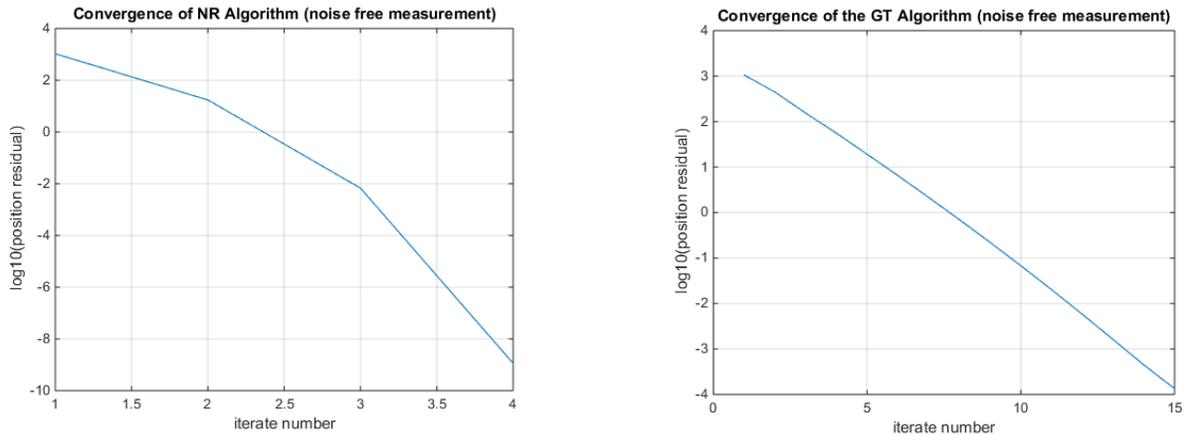

Figure 4. Convergence rate of NR algorithm (left) and GT algorithm (right) under noise free measurement conditions

6. CONCLUSION

In this paper, we perform an in-depth comparison between the GT scheme and the NR method in terms of GPS localization accuracy in the GPS operation environment, its sensitivity with respect to systematic errors and random errors, and computation load required to converge to a localization solution.

While the NR algorithm was shown to have a higher convergence rate than the GT algorithm, the two algorithms operate differently in the information that they require and in the manner in which the solutions are obtained. For example, the computational cost per iteration is greater for the NR algorithm because of the need to invert a matrix at each step of the iteration. In contrast, the GT algorithm requires only one matrix inversion regardless of the number of iterations. When starting with an initial guess which is nearly converged to the final solution, for example from a recent prior position solution, the GT algorithm execution speed is comparable the NR algorithm.

Most spacecraft navigation applications such as the Mars navigation scenario that was considered do not have severe execution speed requirements. For example, many spaceborne GPS receivers have 1 hz position updates which is sufficient for most orbital scenarios. As a result, considerations such as the required navigation infrastructure and error tolerance may be more significant than execution speed. The geometric trilateration algorithm may be seen to be favorable in these other considerations for a navigation scenario at Mars which has limited radionavigation infrastructure.

More fundamentally, the two algorithms process their information in different ways. As was shown in [4], the geometric algorithm may be readily applied to the relative navigation problem using the same measurement processing flow. This may be the most beneficial application of the new algorithm. The Mars navigation application considered in this paper may be recast as a relative navigation algorithm. There are also atmospheric entry, descent, and landing applications which are well-suited to the geometric trilateration problem formulation at Mars.

REFERENCES

[1] http://www.gps.gov/systems/gps/space/

[2] P. Misra and P. Enge, "Global Positioning System: Signals, Measurements, and Performance," 2nd Edition, Ganga-Jamuna Press, 2012.

[3] Irv Leveson, "The Economic Benefits of GPS," http://gpsworld.com/the-economic-benefits-of-gps/.

[4] K.Cheung, C. Lee, "A Trilateration Scheme for Relative Positioning," IEEE Aerospace Conference 2017, Big Sky, Montana, March 2017.

[5] K. Cheung, C. Lee," A Trilateration Scheme for GPS-Style Localization," Interplanetary Network Progress Report, 42-209, May 15, 2017.

[6] K. Cheung, C. Lee, "In-Situ Navigation and Timing Services for the Human Mars Landing Site Part 1: System Concept," accepted for publication in the 68th International Astronautical Congress (IAC), Adelaide, Australia, September 2017.

[7] K. Cheung, C. Lee, G. Lightsey, "In-Situ Navigation and Timing Services for the Human Mars Landing Site Part 2: Modeling, Simulation, and Analysis," abstract submitted to SpaceOps 2018, Marseille, France, May 2018.

BIOGRAPHY

Dr. Kar-Ming Cheung is a Principal Engineer and Technical Group Supervisor in the Communication Architectures and Research Section (332) at JPL. His group supports design and specification of future deep-space and near-Earth communication systems and architectures. Kar-Ming Cheung received NASA's Exceptional Service Medal for his work on Galileo's onboard image compression scheme. Since 1987, he has been with JPL where he is involved in research, development, production, operation, and management of advanced channel coding, source coding, synchronization, image restoration, and communication analysis schemes. He got his B.S.E.E. degree from the University of Michigan, Ann Arbor, in 1984, and his M.S. and Ph.D. degrees from California Institute of Technology in 1985 and 1987, respectively. .

__E. Glenn Lightsey__ is a Professor in the Daniel Guggenheim School of Aerospace Engineering at the Georgia Institute of Technology. He is the Director of the Space Systems Design Lab at Georgia Tech. He previously worked at the University of Texas at Austin and NASA's Goddard Space Flight Center. His research program focuses on the technology of satellites, including: guidance, navigation, and control systems; attitude determination and control; formation flying, satellite swarms, and satellite networks; cooperative control; proximity operations and unmanned spacecraft rendezvous; space based Global Positioning System receivers; radionavigation; visual navigation; propulsion; satellite operations; and space systems engineering. At the University of Texas, he founded and directed the Texas Spacecraft Lab which built university satellites. He has written more than 130 technical publications. He is an AIAA Fellow, and he serves as Associate Editor-in-Chief of the Journal of Small Satellites and Associate Editor of the AIAA Journal of Spacecraft and Rockets.
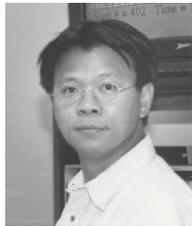
__Charles H. Lee__ is a Professor of Applied Mathematics at the California State University Fullerton and a faculty part-time staff member in the Communications Architectures and Research Section (332) at NASA's Jet Propulsion Laboratory. He received his Doctor of Philosophy degree in Applied Mathematics in 1996 from the University of California at Irvine. Before becoming a faculty member in 1999, he spent three years as a Post-Doctorate fellow at the Center for Research in Scientific Computation, Raleigh, North Carolina, where he was the recipient of the 1997-1999 National Science Foundation Industrial Post-Doctorate Fellowship. He was an Assistant Professor in 1999, an Associated Professor in 2005, and became Full Professor in 2011. His research has been Computational Applied Mathematics with emphases in Control, Fluid Dynamics, Smart Material Structures and Telecommunications, and Biomedical Engineering. He has published over 60 professionally refereed papers, including 25 mathematical and engineering journal articles. In 2002, Dr. Lee received the Outstanding Paper Award from the International Congress on Biological and Medical Engineering.