A Recursive Quasi-optimal Fast Satellite Selection Method for GNSS Receivers

Min Liu, School of Information and Electronics, *Beijing Institute of Technology*
Marc-Antoine Fortin and René Jr Landry, *École de technologie supérieure*

BIOGRAPHY

Min Liu is Ph.D. candidate in the Dept. of Electronic Engineering at Beijing Institute of Technology (BIT). He received his BS from BIT in 2000, and he is currently visiting École de technologie supérieure (ÉTS) in Montréal, Canada, as part of his Ph.D. training. His researches involve GNSS direct-RF signal processing, Compass software receiver and real-time simulator.

Marc-Antoine Fortin is currently pursuing his Ph.D. in the field of GNSS receivers' robustness at École de technologie supérieure (ÉTS) in Montréal, Canada. He previously received an Electrical Engineering Master's degree from École Polytechnique de Montréal (Canada) in 2005 and a Bachelor's degree in the same field from Université de Sherbrooke in 2003 (Canada). He is interested in new methods for universal GNSS acquisition and tracking to be used in severe environments with weak signal levels, multipath and interferences.

René Jr. Landry received a B.Eng. in electrical engineering at the École Polytechnique (Montréal, Canada), in 1992, with a major in space technology. He completed a M.Sc. in satellite communication engineering at the University of Surrey (Guildford, U.K.) in 1993, a Master in space electronics and a DEA in microwaves at SupAéro (Toulouse, France), in 1994. He obtained his Ph.D. degree at University Paul-Sabatier and SupAéro, in 1997, in digital signal processing applied on GPS anti-jamming technologies for the French civil aviation. After a year of post-doctoral fellowship at CNES Toulouse and since 1999, Professor Landry has joined the electrical engineering department of ÉTS. His major interest concerns the development of new mitigation techniques for GNSS receiver robustness in non-ideal environment including cognitive software-defined GNSS receiver, robustness technologies and aided-systems (DGPS, inertial navigation systems combined with communication and new sensor technologies).

ABSTRACT

This paper presents a recursive quasi-optimal satellite selection algorithm for future GNSS receivers to enhance positioning accuracy and robustness with limited resources. For the next decade, new systems, together with existing systems, will provide as much as 40 visible satellites with 160 multi-frequency signals. Any receiver targeting all in-view GNSS signals with fewer channels will need to choose a subset of satellites to be tracked. This is essential for low cost commercial receivers where hardware channels and navigation computer real-time processing capability are limited (hardware/software limitations, complexity of certification, etc.). For example, in order to choose 32 satellites out of 40, traditional algorithms will need more than 50 billion Floating point Operations (FLOPs), which is a huge load for current embedded processors. The algorithm presented in this paper provides quasi-optimal geometry from any combination of signal sources, such as GNSS satellites, space or ground-based augmentation systems, and local constellations. In contrast with the true optimal algorithm, the recursive quasi-optimal algorithm requires a drastically lower computational load. By incorporating various weighting factors, this algorithm can account for User Equivalent Range Error (UERE) when optimizing DOP. Furthermore, when satellites are temporarily blocked, the algorithm provides a list of next-best replacement satellites as a by-product. Receiver Autonomous Integrity Monitoring (RAIM) constraints can also be applied to this algorithm. All these unique features will enhance the robustness and overall accuracy of GNSS receivers. These advantages are unique compared to other quasi-optimal technique of equivalent computational burden.

1 INTRODUCTION

For the next decade, future GNSS systems such as GPS, GLONASS, Galileo and Compass together with existing SBAS systems or even local constellations, will provide more than 40 visible satellites for ground-based users, and even more for space-born users. Figure 1 shows a snapshot of global visibility of all GNSS systems, while Table 1 lists all the GNSS civil signals. These signals will be available on different frequency bands, making the number of visible GNSS signals reach more than 160, in addition to augmentation and local systems' signals.

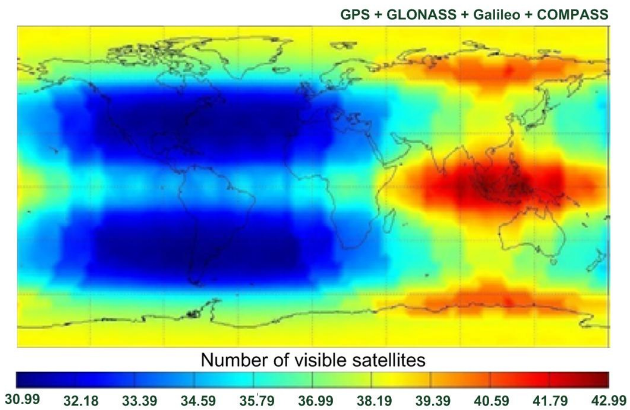

Figure 1: Visibility of All Future GNSS satellites [1]

This will drastically improve ranging systems' availability, continuity, integrity and resistance to interferences, while imposing new challenges on the receiver side. Given a 40-channel multi-frequency receiver, several channels may be used for the same satellite, thus leaving a large number of untracked visible satellites. Despite the fact that "all-in-view" receivers may avoid this issue by accommodating all the visible signals, the optimum performance with the lowest number of channels is always preferable in terms of system complexity, power consumption and cost, especially for most commercial devices with limited number of channels (ex. 24 to 48). Hence, it is highly desirable to have a fast satellite selection method to choose the ideal subset of visible satellites in order to continuously maintain the best satellites' configuration, in real-time.

Considering all current and future civil GNSS signals, performing a true optimal satellite selection will impose a huge computational load, especially as visible satellites grow in number. For example, choosing 20 satellites out of 40 represents approximately $1x10^{11}$ combinations. The optimal algorithm calculates each of their Dilution of Precision (DOP), or any other criterion, and finds the combination producing the best results. In this case, this technique is only possible in post-processing software. Moreover, even with a small number of visible satellites, the recalculation rate needs to be fast enough to prevent out-of-date satellite combinations from degrading the overall navigation accuracy.

Table 1: Current and Future GNSS Civil Signals

System	# of satellites	Band	Civil Signal
GPS	32	L1	C/A
	32		L1C (I/Q)
	32	L2	L2C (CM/CL)
	32	L5	L5 (I/Q)
Galileo	30	E1	E1 (B/C)
	30	E5	E5a (I/Q)
	30		E5b (I/Q)
GLONASS	24	L1	L1OF
	24		L1ROC
	24	L2	L2OF
	24	L5	L5OF
	24		L3ROC
Compass	35	E1-1	B1-1: C/A (I)
	35	E5B	B2: C/A (I)
	35	E6	B3: C/A (I)

This paper presents a new satellite selection algorithm for future GNSS receivers to produce more accurate Position, Velocity and Time (PVT) estimates by choosing the optimal subset of GNSS satellites. Knowing all the almanacs and an initial receiver PVT, the proposed recursive quasi-optimum algorithm gives DOP estimates only one percent larger in average compared to the all-combinations true optimum method, but with dramatically lower computational load. This algorithm can assess satellite geometry targeting any DOP metric, so that it could be flexibly tailored to any specific application and easily changed to match any user dynamic requirements. For example, a terrestrial user could use GDOP; a spacecraft receiver equipped with precise clock should target Position DOP (PDOP) while the Vertical DOP (VDOP) might be of greater interest for space shuttles' touchdown and airplane landing. As a by-product, it generates a sequence of next-best satellites for immediate replacement in case of signal blockage/outage, multipath or other strong signal perturbations. Thus, during navigation, the DOP will not noticeably increase if these replacement satellites are immediately tracked and used in the navigation solution upon loss of one or more satellites (i.e. before the next iteration of this algorithm). RAIM constraints can also be inserted into the proposed algorithm, screening out inadmissible satellites even if they would provide the best metrics. Furthermore, various weighting factors can be incorporated into the satellites' selection to account for measurement discrepancies and to minimize errors. The weighting factors are formed by proper combination of Signal-to-Noise Ratio (SNR), User Range Accuracy (URA), atmospheric modeling error, and Kalman filter error estimates. Hence, with near-optimal geometry, low computational load and various features, the recursive quasi-optimal algorithm is well suited for real-time applications that pursue future multi-frequency GNSS constellations with greater accuracy, reliability and flexibility.

This paper first reviews the existing satellite selection methods. Then, it presents the basic idea behind the proposed recursive quasi-optimal Satellite Selection algorithm and its implementation details, including algorithm optimizations and generation of fast-replacement backup satellites list. It is followed by a performance comparison with other existing methods using both random and simulated constellations by Monte-Carlo analysis. The computational load is assessed, and different weighing factors are summarized. Next, RAIM-admissible constraints are further discussed, and the potential to incorporate them in the proposed algorithm is demonstrated, making the receiver RAIM-ready. Finally, the optimal geometry's changing rate is studied and the impacts of the iteration rate on the overall DOP loss are presented. A brief summary ends this paper with the advantages of the proposed technique.

2 NAVIGATION ERROR MODEL

Early GPS receivers with 4 to 6 channels needed to select an optimum subset from all the visible satellites (i.e. 8 to 12). Nowadays, this problem is bypassed by most of the all-in-view receivers. However, for the new multi-constellation multi-frequency GNSS era, satellite selection is an old problem with a new flavor, and it will impose new challenges on GNSS receivers' design.

Position and time estimates' accuracy is one of the most important performance metrics for GNSS receivers. The standard deviation of navigation error is modeled as [2]:

$$\sigma_{3D\ position\ \&\ Time} = \sigma_{UERE} * GDOP \tag{1}$$

Where σ_{UERE} is the standard deviation of User Equivalent Range Error common to each satellite and GDOP stands for Geometric Dilution Of Precision. The UERE is the difference between the true and measured user-satellite distances, and is further divided into Signal In Space User Range Error (SIS URE) and User Equipment Errors (UEE) [3]. The GDOP describes the user-satellites geometric relationship, which greatly affects navigation accuracy. It is possible to model other position errors by using corresponding DOPs: for example, the position error is modeled using PDOP:

$$\sigma_{3D\ position} = \sigma_{UERE} * PDOP \tag{2}$$

Because the UERE is generally different for all satellites [4], a minimum GDOP cannot guarantee minimum navigation error. For example, a satellite at low elevation angle might provide optimal geometry but the pseudorange error can be significantly larger than those of other satellites. Nonetheless, to have a minimum GDOP is always the first step in achieving optimum navigation performances. Also, it is possible to incorporate various weighting factors to account for UERE discrepancies in the satellite selection process, thus producing a set of satellites that has minimum weighted GDOP. This technique is further discussed in section 4.4.

3 SATELLITE SELECTION ALGORITHMS

The best position solution for a given measurement error variance is the one that minimizes GDOP. GDOP, in turn, is minimized for an assumed user location by optimizing the relative geometry of the selected satellites [1]. There are generally three kinds of algorithms to minimize DOP. The optimal selection algorithm yields the true minimal DOP at the cost of a high computational load. Sub-optimal algorithms greatly reduce the computational load, but provide reasonably higher DOPs. Furthermore, it should be noted that selecting more than four satellites is possible, although algorithms are optimized for four satellites' selection. The addition of more satellites is based on additional DOP, whose computational load is high [8]. Most excitingly, a quasi-optimal selection algorithm is introduced in section 3.3; it produces quasi-optimal DOP with low computational load.

3.1 OPTIMAL SATELLITE SELECTION

Given N visible satellites, there are $C_k^N = \frac{N!}{k!\,(N-k)!}$ possible k–subsets, each comprised of a unique combination of k satellites. By computing the DOP of each subset, and searching for the resulting minimal DOP, the optimal satellite combination is selected. For example, to choose 6 satellites out of 12, DOP calculation and comparison would be applied $C_6^{12} = 924$ times. This might be possible for ground-based GPS receivers, whose maximum visible satellites will never exceed 12. On the other hand, for future GNSS receivers, choosing 20 satellites out of 40 implies reproducing $C_{20}^{40} = 1.38 \times 10^{11}$ times the same computations. As processors become more powerful, this load may be achievable in real-time. In this case, real-time requirements depend on how fast the satellites' geometry changes, as further described in Section 5. Indeed, a single GDOP calculation is defined as [4]:

$$GDOP = \sqrt{trace[(G^T G)^{-1}]} \tag{3}$$

$$G = \begin{bmatrix} \mathrm{los}_1 & 1 \\ \mathrm{los}_2 & 1 \\ \vdots & \\ \mathrm{los}_k & 1 \end{bmatrix} \tag{4}$$

where G is the direction cosine matrix and its row vector is composed by the line-of-sight vector los_i from the receiver to the i^{th} satellite, and a constant 1. This requires 4x4 matrix multiplication and inversion, both of which are processor intensive operations. When executed in a loop, whose count rapidly increases with the number of visible satellites, the optimal algorithm consumes huge FLOPs, as shown in figure 2.

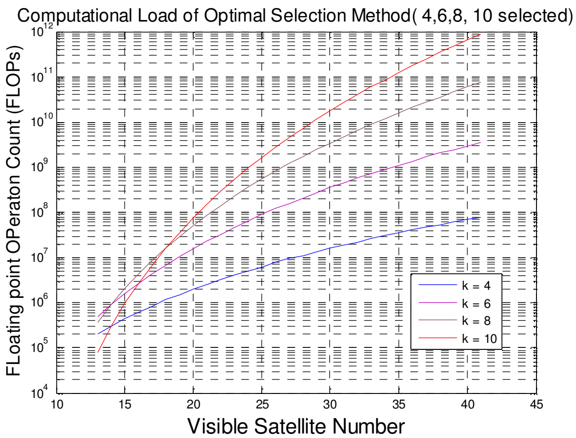

Figure 2: Computational Load of Optimal Satellite Selection (4, 6, 8, 10 SVs selected out of 13 to 41)

3.2 SUB-OPTIMAL SATELLITE SELECTION ALGORITHMS

The most popular satellite selection algorithm was developed by Dr. W. M. Lear, called "Lear's simple satellite selection algorithm" [5] or more widely known as the "Highest Elevation" algorithm. It first selects the satellite of largest elevation angle. Then, new satellites are added on a best-match basis: the second is 90° away from the first; the third is perpendicular to the plane formed by the two first; the fourth optimizes GDOP, and the fifth minimizes PDOP. It produces reasonably good results with a very small computational load. Note that it is possible to extend this algorithm to select more than five satellites.

When selecting four satellites, the selected satellites will form a tetrahedron with the receiver, whose volume is said to be approximately inversely proportional the corresponding GDOP. Hence, by maximizing the volume, the GDOP is minimized [6]. Various algorithms are developed based on this theory. Instead of computing GDOP, the most straightforward way calculates the volume of the tetrahedron of all four-satellite combinations. Other variants select the first satellite with the largest elevation angle [6] or along the velocity vector [5]; the second step for both techniques targets a satellite 109.5° away from the first; the third and fourth satellites maximize the volume of the tetrahedron. A third variant based on the tetrahedron theory is developed by Li [7], and referred to by [8] as the "Four-step algorithm", which yields sub-optimal geometry with reduced computational load. It first selects the highest elevation angle satellite; the second has the largest angle distance with the first; the third satellite is selected for its smallest angular distance to either of the two still unpopulated vertices of the tetrahedron; and the fourth is selected to minimize PDOP. It is also possible to further select more satellites based on DOPs. However, the resultant DOP will not decrease significantly due to the algorithm's limitations.

Extending the 4-satellite optimal geometry theory, Zhang et al. [9] try to find new optimal geometries for greater subsets. They developed a fast satellite selection algorithm to identify the satellites most similar to these optimal geometries. The geometries found for 5 to 16 satellites share the same characteristic: two to five satellites lie at zenith while others are uniformly distributed at the horizon circle. Based on these pre-calculated optimal geometries, the number of satellites to be tracked at zenith is determined for any number of selected satellites. The satellite with smallest elevation angle is also selected, and the remaining visible satellites are grouped and screened out by comparing their azimuth angles with this satellite against a threshold. Simulation results show that there is an average of 5.79% GDOP increase compared with the optimal algorithm, and a worst case of 50% GDOP increase.

3.3 QUASI-OPTIMAL SATELLITE SELECTION ALGORITHM

A quasi-optimal satellite selection algorithm is introduced in [8] by MIT researchers: it produces a selected subset whose PDOP is nearly 100% the optimal one. Most differences are within 5% of the optimal PDOP, and the worst case deviation reaches 23% using random constellations, but only 4% higher using simulated constellations. This quasi-optimal algorithm is based on the intuition that the line-of-sight vectors that are co-linear are redundant. This inspires the design of a cost function:

$$J_{i,j} = cos2\theta_{i,j} \tag{5}$$

where $\theta_{i,j}$ is the angle between los_i and los_j so that the cost is highest when the two vectors are co-linear ($\theta_{i,j} \approx 0°\ or\ \theta_{i,j} \approx \pm 180°$) and lowest when they are perpendicular ($\theta_{i,j} \approx 90°$). The cost for the i^{th} satellite is the sum of all the cost functions of this satellite to all the *N* visible satellites:

$$J_i = \sum_{j=1}^{N} cos2\theta_{i,j} \tag{6}$$

Without introducing the optimization details, this quasi-optimal algorithm executes as follows:

1. Computes the cost for all visible satellites;
2. Eliminates the satellite having the largest cost, which is the most redundant satellite and provides the least additional information;
3. If the remaining satellites number is greater than desired, repeat 1 and 2.

Thanks to the simplified design of the cost function in (5), equation (6) is very easy to compute from the direction cosine matrix and various optimizations can be applied to significantly reduce the computational load. The FLOPs needed to select 6 satellites out of 7 to 15 is shown in Figure 3 [8].

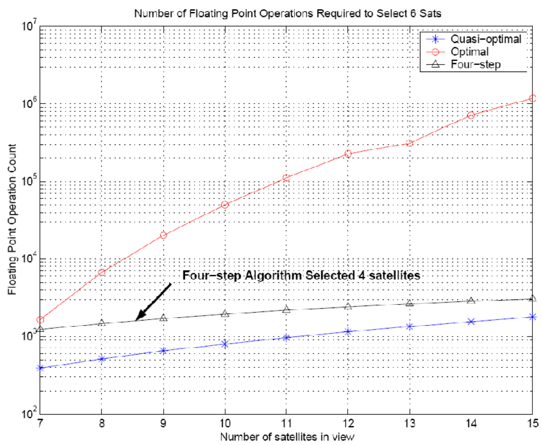

Figure 3: Computational Load of Optimal Satellite Selection (6 SVs selected out of 7 to 15) [8]

It can be seen that the quasi-optimal algorithm is more suitable for real-time applications. The proposed recursive quasi-optimal satellite selection algorithm will be introduced in the next section.

4 PROPOSED RECURSIVE QUASI-OPTIMAL SATELLITE SELECTION

This section presents a newly addition to the quasi-optimal algorithm family: the proposed recursive quasi-optimal satellite selection algorithm. It yields quasi-optimal constellation geometries with any desired number of satellites. The proposed method matches and outperforms in some cases the existing MIT quasi-optimal algorithm in terms of DOP error with a small increase in computational load.

4.1 RECURSIVE QUASI-OPTIMAL ALGORITHM DESIGN

The proposed recursive satellite selection algorithm is derived from the following phenomenon:

> For any given *N* visible satellites, a *k*–subset (containing *k* satellites) providing optimal geometry tends to share most elements with a *(k-1)* –subset of optimal geometry.

This can be verified by simulation using randomly generated direction cosine matrix or a simulated single- or multi-systems satellites' constellation. Table 2 shows an example of a simulated GPS-only constellation using Spirent SimGen software logged data.

Table 2: Selected k Satellites out of 13 Visible Satellites

k	Selected SVs out of 13 Visible Ones												
12		2	3	4	5	6	7	8	9	10	11	12	13
11		2	3	4		6	7	8	9	10	11	12	13
10		2	3	4			7	8	9	10	11	12	13
9		2		4			7	8	9	10	11	12	13
8		2		4			7	8	9	10	11	12	
7		2	3	4				8	9		11	12	
6		2		4					9	10	11	12	
5	1			4		6			9		11		
4		2		4			7				11		

A total of 13 visible satellites are generated and k ($k\epsilon[4,12]$) selected satellites using optimal satellite selection algorithm. Comparing two successive rows from top to bottom, each new row will drop one satellite from k=12 to k=8, and maintain the same remaining satellites. As k decreases to smaller numbers, newly selected satellites are highlighted by colored cells. The successive subsets' differences being small (especially for larger *N*), it becomes possible to select *k-1* satellites out of the previously selected *k* satellites without noticeably degrading the solution. This selection process is applied recursively to reduce *k* from the total visible satellites' number to the desired number of satellites to be tracked.

Given *N* visible satellites, using this recursive algorithm to select *M* (*M*<*N*) satellites runs as follows:

1. Initialize k = N;
2. Generate all (*k-1*)–subsets (i.e. each (*k-1*)–subset has *k-1* satellites), the total number of subsets is $C_{k-1}^{k} = k$;
3. Compute GDOP or (any other metric) for each (*k-1*) –subset;
4. Find the smallest GDOP within the *k* results;
5. Put the satellite excluded from the selected (*k-1*) –subset to a list for fast replacement;
6. Remove the rejected satellite and decrease *k*;
7. Return to Step 2 until *k* equals *M*.

Notably, Step 2 is simply excluding satellite 1 to satellite *k* for each (*k-1*) – subset. The most computational intensive processing is Step 3, which calculates *k* different GDOPs (i.e. one for each subset), involving matrix multiplication and inversion, both of which can be optimized due to similarities between any two subsets. From all the *k* line-of-sight vectors, a single direction cosine matrix is formed in (3), where the i^{th} line-of-sight vector can be formulated as:

$$los_i = [(\mathrm{x_i}, \mathrm{y_i}, \mathrm{z_i}), 1], 1 \leq i \leq k \quad (7)$$

Calculating the matrix multiplication in (8) is required only once at the first iteration of the satellite selection process (when k=N) since all subsequent matrix multiplications can be derived from it. The matrix multiplication of N satellites leading to the original reference intermediate result Q_N is:

$$Q_N = G^T G = \begin{bmatrix} \sum_{t=1}^{N} x_t x_t & \sum_{t=1}^{N} x_t y_t & \sum_{t=1}^{N} x_t z_t & \sum_{t=1}^{N} x_t \\ \sum_{t=1}^{N} y_t x_t & \sum_{t=1}^{N} y_t y_t & \sum_{t=1}^{N} y_t z_t & \sum_{t=1}^{N} y_t \\ \sum_{t=1}^{N} z_t x_t & \sum_{t=1}^{N} z_t y_t & \sum_{t=1}^{N} z_t z_t & \sum_{t=1}^{N} z_t \\ \sum_{t=1}^{N} x_t & \sum_{t=1}^{N} y_t & \sum_{t=1}^{N} z_t & N \end{bmatrix} \quad (8)$$

Then, for each iteration, the intermediate results $Q_{k-1,i}$ (i.e. k-1 satellites for any excluded SV_i) are derived from the previously selected Q_k :

$$Q_{k-1,i} = Q_k - \begin{bmatrix} x_i x_i & x_i y_i & x_i z_i & x_i \\ y_i x_i & y_i y_i & y_i z_i & y_i \\ z_i x_i & z_i y_i & z_i z_i & z_i \\ x_i & y_i & z_i & 1 \end{bmatrix} = Q_k - {K_i}^T K_i \quad (9)$$

$$K_i = [los_i \; 1] \quad 1 \le i \le k \quad (10)$$

where K_i is the extended line-of-sight vector. Thus, by simply saving the Q_k of the selected subset, all matrix multiplications of the next iterations can be calculated by (9).

According to (3), the inversion of matrix Q is needed for the GDOP computation, and this can be done using the Matrix Inversion Lemma introduced in [10]. This method takes advantage of the minimal changes of any two adjacent subsets and calculates matrix inversion in an iterative approach. This algorithm can produce any (k-1) – subset matrix inversion from the previously selected k– subset matrix inversion. This approach is approximately 40% faster than the LU Decomposition method for 4x4 matrices, according to the Author's estimation by summing all the arithmetical operations for both methods.

Note that initially, the inversion of Q_N is computed using LU Decomposition. Then for each subset excluding the i^{th} satelite, inversion of $Q_{k-1,i}$ is carried out and $GDOP^2$ is calculated as follows:

1. $K_i = [los_i \; 1] \qquad 1 \le i \le N$
2. $a = K_i * Q_k$
3. $b = \frac{a^T}{a * K_i^T - 1}$
4. $Q_{k-1,i} = Q_k - b * a$
5. $GDOP_{k-1,i}^2 = trace(Q_{k-1,i})$

Then, find the minimum among the k combinations of $GDOP_{k-1,i}^2$, and update Q_k with the corresponding $Q_{k-1,i}$. This completes one iteration of the satellite selection process.

At each iteration, one satellite is rejected and stored as a fast-replacement satellite. Signal blockage being common for receivers moving in an urban canyon environment, fast-replacement backups provide next-best satellites to track in case of satellite losses before the next calculation of the quasi-optimal satellite subset. To select k satellites out of N, the proposed algorithm will need N-k iterations. Each iteration will exclude one satellite and this satellite is pushed into a stack. When needed, pop out the latest backup satellite from the stack's top; this latest backup satellite being, after all, the one of the former selected quasi-optimal (k+1)–subset. The replenished k–subset should have GDOP no worse than the largest GDOP of the $k+1$ subsets. This is further proved by simulation and demonstrated in the following section 4.2 after the performances' evaluation of the proposed recursive quasi-optimal satellite selection algorithm.

4.2 PERFORMANCE EVALUATION

This section presents a performance comparison of the proposed approach with the MIT quasi-optimal method by Monte-Carlo analysis using both random and simulated constellations. Random constellations are more general and give overall performance insights of the algorithms. Simulated constellations are used to validate the proposed algorithm, and provide more practical results. It is not necessary to compare the proposed algorithm with other sub-optimal methods, as the MIT algorithm presented in [8] is much superior than those algorithms. The optimality is evaluated using the PDOP's ratio of the quasi-optimal to the optimal methods [8], defined as

$$\zeta_{quasi} = \frac{PDOP_{quasi}}{PDOP_{optimal}} \quad (11)$$

Since the optimal algorithm yields the smallest PDOP from given N satellites, the magnitude of ζ_{quasi} is always greater (or equal) than 1, and the closer to 1, the better performance of the algorithm being evaluated. Although the recursive quasi-optimal algorithm can use any DOP, the PDOP is first used as the geometry metric, since the MIT quasi-optimal algorithm is not designed to optimize GDOP. After adapting the MIT algorithm, the comparison is then extended to GDOP.

4.2.1 SATELLITE SELECTION FROM RANDOM CONSTELLATIONS

A total of 1000 random direction cosine matrices were generated in Matlab. The MIT quasi-optimal algorithm, the proposed recursive quasi-optimal algorithm and the optimal algorithm were applied. Both quasi-optimal techniques' results were separately evaluated against the results of the optimal satellite selection method. The statistical distribution of $\zeta_{quasi-MIT}$ and $\zeta_{quasi-recursive}$ are given in Figure 4 and Figure 5.

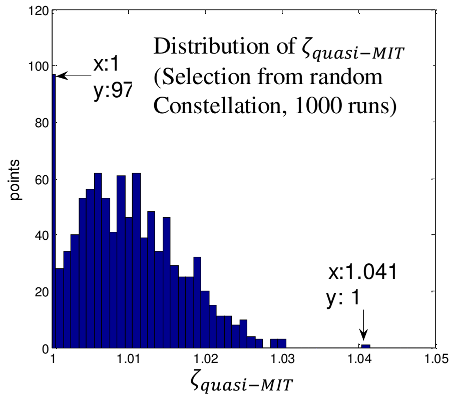

Figure 4: Distribution of $\zeta_{\mathrm{quasi-MIT}}$(PDOP)

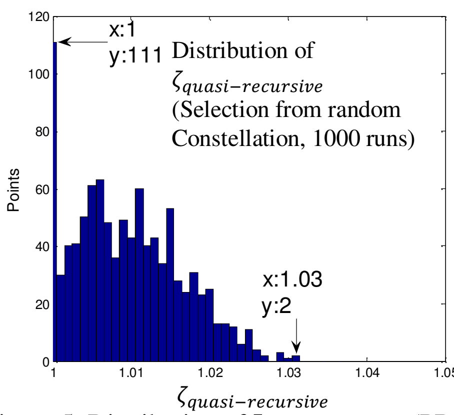

Figure 5: Distribution of $\zeta_{\mathrm{quasi-recursive}}$(PDOP)

Both algorithms perform exceptionally well: the recursive quasi-optimal algorithm has 111 cases out of 1000 runs that achieve the optimal PDOP, while MIT algorithm only achieves 97 such cases. At the other end of the spectrum, two runs out of 1000 of the proposed recursive algorithm reach a maximum of 3% deviation, while the MIT algorithm's maximum point exceeds the optimal PDOP by a 4.1% deviation.

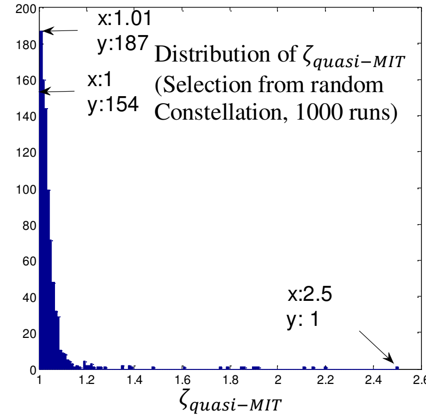

Figure 6: Distribution of $\zeta_{\mathrm{quasi-MIT}}$(GDOP)

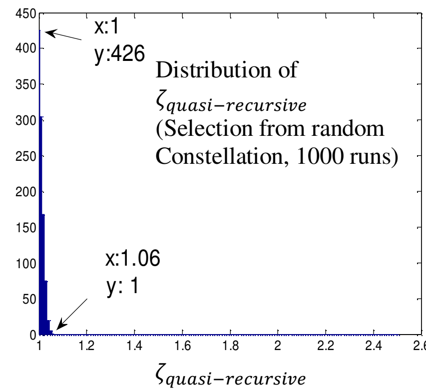

Figure 7: Distribution of $\zeta_{\mathrm{quasi-recursive}}$(GDOP)

The same simulation was also conducted with the GDOP metric. Corresponding results are shown in Figure 6 and Figure 7. It can be seen that the proposed recursive algorithm outperforms the MIT solution with 42.6% $\zeta_{quasi-recursive}$ of cases matching the optimal solution compared to only 18.7% that of the $\zeta_{quasi-MIT}$. Furthermore, the maximum deviation of the proposed algorithm shows 6% larger GDOP than the optimal one in only one occasion out of 1000. On the other hand, the maximum deviation of MIT's algorithm reaches 2.5 times the optimal GDOP.

4.2.2 SATELLITE SELECTION FROM SIMULATED GPS CONSTELLATIONS

Simulated GPS constellations were generated using the Spirent SimGen simulation software. The simulated satellites' position was recorded using the built-in bulk logging function of SimGen. The logging interval was set to 6 minutes, and a total of 2 days' data were collected. The user was static at coordinates N40 W80 and was set to a high elevation of 80km, so that the visible satellites could reach 13. A total of 576 samples (9.6-hour long) were tested using the three algorithms applied from 4 to 9 satellites. More satellites and longer data could have been generated and used, but it would have required weeks of calculations due to the optimal algorithm's extremely long computation time. The mean and maximum values of $\zeta_{quasi-MIT}$ and $\zeta_{quasi-recursive}$ are summed up in Table 3 and Table 4, respectively. Note that the proposed approach may not perform as well with very low (i.e. 4-5) satellites number.

Table 3: Mean value of $\zeta_{quasi-MIT}$ and $\zeta_{quasi-recursive}$

SV Selected	4	5	6	7	8	9
Mean $\zeta_{quasi\text{-}MIT}$	1.024	1.014	1.008	1.011	1.014	1.017
Mean $\zeta_{quasi\text{-}ecursive}$	1.024	1.014	1.008	1.011	1.013	1.017

Table 4: Maximum value of $\zeta_{quasi-MIT}$ and $\zeta_{quasi-recursive}$

SV Selected	4	5	6	7	8	9
Max $\zeta_{quasi\text{-}MIT}$	1.070	1.052	1.033	1.031	1.041	1.051
Max $\zeta_{quasi\text{-}ecursive}$	1.077	1.040	1.031	1.029	1.041	1.051

From the statistics of the 576 runs, the recursive quasi-optimal selection algorithm slightly outperforms the MIT quasi-optimal algorithm, especially in terms of the max deviation. Both algorithms are, on average, less than 3% greater than the optimal results.

4.2.3 FAST REPLACEMENT EVALUATION

Worst case analysis of fast replacement satellites was simulated using randomly generated direction cosine matrices. Given *N* visible satellites, the recursive quasi-optimal algorithm selects the *k*–subset that has the greatest PDOP out of *k+1* satellites, thus the rejected one is regarded as lost. Then, an optimal satellite selection is applied to *k-1* satellites (excluding the dropped satellite). The PDOPs of the optimal selection results are compared with that of the recursive quasi-optimal selection for $N = 12$, and $k = 6$ to 10. The distribution of $\zeta_{fast\ replace}$ defined in (12) is collected during 1000 runs.

$$\zeta_{fast\ replace} = \frac{PDOP_{worst\ case}}{PDOP_{optimal}} \qquad (12)$$

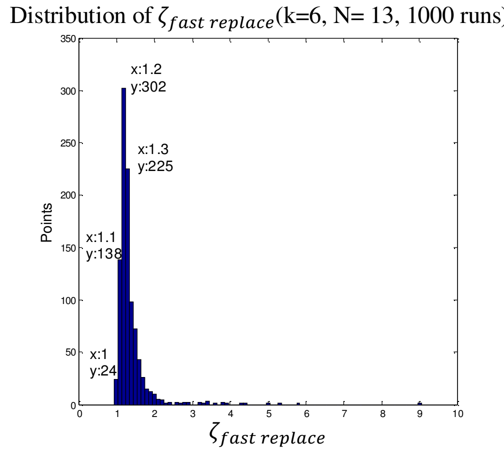

Figure 8: Distribution of 6 satellites

In Figure 8, the first 4 bins added together account for 68.9% of the worst case PDOP; using fast replace satellite has a 30% growth in PDOP. Maximum deviation is 8.9 times the optimal result, but by rare chance (1 out of 1000). The statistical results are listed in Table 5. It can be seen that the worst case mean value of $\zeta_{fast\ replace}$ is reasonably good from 1.37 to 1.08, and the maximum deviation decreases as selected number of satellites grows.

Table 5: Statistics of $\zeta_{\text{fast replace}}$ for 6 -10 SVs out of 13

SV Selected	6	7	8	9	10
Mean $\zeta_{fast\ replace}$	1.373	1.211	1.155	1.123	1.082
Max $\zeta_{fast\ replace}$	8.951	4.124	2.684	2.255	1.72

4.3 COMPUTATIONAL LOAD

The computational load for selecting 12 satellites out of 13 to 45 in-view satellites is estimated for optimal, MIT quasi-optimal and recursive quasi-optimal selection algorithms and shown in Figure 9. In order to comply with the MIT algorithm, the metric used in the recursive quasi-optimal algorithm is PDOP, thus 3x3 matrix inversions are performed. The computational load is evaluated by FLOPs, where each float point addition, multiplication and division is regarded as one FLOP.

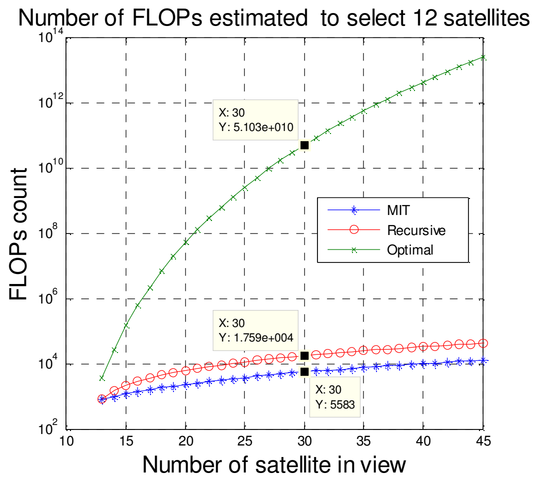

Figure 9: Computational Loads' Comparison

As one can observed from Figure 9, the computational load for the recursive quasi-optimal method is approximately two to three times higher than the MIT quasi-optimal algorithm, but still six orders of magnitude lower than the optimal algorithm applied to 30 in-view satellites. Also notice that the optimal algorithm grows by ten orders of magnitude from 13 to 45 satellites, while the recursive quasi-optimal algorithm only increases by 50 times.

4.4 WEIGHTED RECURSIVE QUASI OPTIMAL ALGORITHM

The error model in (1) suggests that both the geometry and the UERE should be kept as small as possible to enhance the navigational accuracy. During the satellite selection process, it is desirable to account for the two factors. In order to do so, weighted satellite selection algorithms provide possible solutions. Research has been done on the topic: by referring to User Ranging Accuracy (URA) and to the satellite health information in the navigation data [11], by modeling iono- and tropospheric delays based on the elevation angle [12], by considering signal-noise-ratio [13], by evaluating the received power by the satellite-user distance [14] or by evaluating Kalman filter's residuals, a weighing matrix W can be formed and applied to the recursive quasi-optimal satellite selection algorithm. Thus (3) becomes:

$$GDOP = \sqrt{trace[(G^T W G)^{-1}]} \tag{13}$$

$$W = \begin{bmatrix} \mathrm{w}_1 & & \\ & \ddots & \\ & & \mathrm{w}_k \end{bmatrix} \tag{14}$$

And (9) extends to

$$\begin{aligned} Q_{k-1,i}^{w} &= Q_k^w - \mathrm{w}_i \begin{bmatrix} x_i x_i & x_i y_i & x_i z_i & x_i \\ y_i x_i & y_i y_i & y_i z_i & y_i \\ z_i x_i & z_i y_i & z_i z_i & z_i \\ x_i & y_i & z_i & 1 \end{bmatrix} \\ &= Q_k - \mathrm{w}_i {K_i}^T K_i \end{aligned} \tag{15}$$

The weighting factors can also be used to reduce the number of satellite transitions, which tend to cause slow convergence of the carrier phase ambiguity states and to degrade state estimates. For some applications, weighting factors can apply extra strength favoring rising satellites in order to have longer carrier phase accumulation.

4.5 RAIM-ENABLED SATELLITE SELECTION

For some applications, robustness is of great concern, and Receiver Autonomous Integrity Monitoring (RAIM) algorithms might be employed to detect and exclude potential unexpected satellites' faults. These kinds of algorithm use the redundancy from at least five satellites, imposing a constraint on the satellite selection, which makes it possible for RAIM or Fault Detection and Exclusion (FDE) algorithms to meet designed false alarm rate and detection probability. Otherwise, the selected satellite geometry is referred to as being inadmissible [4] even if it yields very good DOP. Indeed, inadmissible geometries lack the appropriate redundancy for integrity check and could be screened out at satellite selection stage. The integrity constraint is formulated in [10] as:

$$\delta H_i^2 = DOP_i^2 - DOP^2, i = 1, .., k \tag{16}$$

$$\max \delta H_i^2 \le \delta H_{Thresh}^2, \forall i \in k \tag{17}$$

where DOP is the DOP of k selected satellites and DOP_i is the DOP of k-1 satellites with the i^{th} satellite removed. δH_{Thresh}^2 is a threshold predetermined for Probability of False Alarm (PFA) and Probability of Missed Detection (PMD) to meet the worst case fault detection requirement. This constraint can be easily incorporated into the recursive quasi-optimal satellite selection algorithm and hence make the selected satellites RAIM-admissible. Simply by running one more iteration of the recursive algorithm and calculate $k\ \delta H_i^2$ from the k satellites, and comparing to the predetermined threshold, the constraint is met when (17) is satisfied for all i, or a constraint violation is reported and further actions may be taken.

5 IMPACTS OF OPTIMAL GEOMETRY CHANGING RATE ON SATELLITE SELECTION PERFORMANCES

As the receiver roams and GNSS satellites orbit the Earth, the subset yielding optimal geometry is continuously changing. Thus, satellite selection needs to be executed periodically to avoid out-of-date satellite geometries degrading the navigation solution's accuracy. The optimal geometry changing rate dictates the satellite selection's ideal repeat interval, especially for low dynamic receivers [5]. This changing rate was studied by simulation assuming static receiver at N39 E72, on sea level. A 48-hour data set was generated by SimGen with a one minute resolution, to which the optimal satellite selection algorithm was applied. The interval between two successive optimal subsets is analyzed and its statistics are given in Table 6 for selecting 4 to 8 satellites out of 12. The optimal geometry changes faster as the receiver moves, thus the mean value in Table 6 can be regarded as worst case bound and the selection repeat interval should be no longer than those values.

Table 6: Optimal Geometry Changing Rate

SV Selected	4	5	6	7	8
Min Interval (min)	1	1	1	1	1
Max Interval (min)	66	73	58	45	101
Mean Interval (min)	12.4	11.3	12.3	12.6	15.7

6 SUMMARY

In conclusion, the proposed recursive quasi-optimal satellite selection algorithm yields near-optimal geometry at a very low computational cost. It can target any DOP metrics, while weighting factors can account for UERE errors during satellite selection. A stack of fast replacement backup satellites is generated as a by-product of the proposed algorithm. RAIM constraints can be easily incorporated by running an extra recursive selection iteration to check the selected satellites' admissibility to fault detection algorithms. The proposed algorithm outperforms all the existing fast satellite selection algorithms. Such performances, flexibility and versatility make the recursive quasi-optimal satellite selection algorithm ideal for GNSS receivers' real-time applications.

ACKNOWLEDGMENTS

M. Liu is funded by the China Scholarship Council as a visiting scholar in ÉTS.

M.-A. Fortin is financially supported by Averna Technologies, the *Fonds Québécois de Recherche sur la Nature et les Technologies* (FQRNT) and the Natural Sciences and Engineering Research Council of Canada (NSERC) in the form of a BMP industrial innovation scholarship.

REFERENCES

[1] T. Meng-Lun, H. Yu-Sheng, C. Kai-Wei, and Y. Ming, "The Impact of Compass/Beidou-2 on Future GNSS: A Perspective from Asia," in *ASC2008* Taiwan, 2008.

[2] E. H. Martin, "GPS User Equipment Error Models," in *Journal of Navigation*. vol. 25 Washington, DC: Institute of Navigation, 1978, pp. 201-210.

[3] P. Misra and P. Enge, *Global Positioning System: Signals, Measurements, and Performance*, 2nd edition ed. Lincoln, Massachussetts: Ganga-Jamuna Press, 2006.

[4] J. J. Spilker and B. W. Parkinson, *Global Positioning System: Theory and Applications* vol. 1: AAIA, 1996.

[5] S. P. Cryan, M. Douglas, and M. N. Montez, "A Survey of GPS Satellite Selection Algorithms for Space Shuttle Autolanding," in *Fifth International Technical Meeting of The Satellite Division of the Institute of Navigation*, 1992, pp. 1165-1171.

[6] M. K. a. T. Okada, "A SATELLITE SELECTION METHOD AND ACCURACY FOR THE GLOBAL POSITIONING SYSTEM," *Journal of Navigation,* vol. 31, pp. 8-20, 1984.

[7] J. Li, A. Ndili, S. Buchman, and L. Ward, "GPS RECEIVER SATELLITE/ANTENNA SELECTION ALGORITHM FOR THE STANFORD GRAVITY PROBE B RELATIVITY MISSION," in *1999 National Technical Meeting Proceedings of the Institute of Navigation*, 1999, pp. 541 - 550.

[8] P. C.W. and J. P. How, "QUASI-OPTIMAL SATELLITE SELECTION ALGORITHM FOR REAL-TIME APPLICATIONS:," in *Proceedings of the 14th International Technical Meeting of the Satellite Division of the Institute of Navigation ION GPS 2001*, 2001, pp. 3018-3028.

[9] M. Zhang, J. Zhang, and Y. Qin, "Satellite selection for multi-constellation," in *IEEE PLANS* Piscataway, NJ 08855-1331, United States: Institute of Electrical and Electronics Engineers Inc., 2008, pp. 1053-1059.

[10] M. S. Phatak, "Recursive method for optimum GPS satellite selection," *IEEE Transactions on Aerospace and Electronic Systems,* vol. 37, pp. 751-4, 2001.

[11] J. W. L. a. C. Shank, "A Study into the Use of URA in Satellite Selection," in *Proceedings of the 49th Annual Meeting of the Institute of Navigation*, 1993, pp. 261 - 270.

[12] P. Chansik, K. Ilsun, L. Jang Gyu, and J. Gyu-In, "A satellite selection criterion incorporating the effect of elevation angle in GPS positioning,"

Control Engineering Practice, vol. 4, pp. 1741-6, 1996.

[13] Y. Yang and L. Miao, "GDOP results in all-in-view positioning and in four optimum satellites positioning with GPS PRN codes ranging," in *PLANS 2004. Position Location and Navigation Symposium* Piscataway, NJ, USA: IEEE, 2004, pp. 723-7.

[14] M. C. Moreau, "GPS receiver architecture for autonomous navigation in high earth orbits." vol. Ph.D United States -- Colorado: University of Colorado at Boulder, 2001, p. 203.