Exploiting the secondary codes to improve signal acquisition performance in Galileo receivers.

Author:
SHIVARAMAIAH, N.C; DEMPSTER, A.G; RIZOS, C

Publication details:
Proceedings of the 21st International Technical Meeting of the Satellite Division of the Institute of Navigation ION GNSS 2008
pp. 1497-1506

Event details:
21st Int. Tech. Meeting of the Satellite Division of the U.S. Inst. of Navigation
Savannah, USA

Publication Date:
2008

DOI:
https://doi.org/10.26190/unsworks/725

Exploiting the Secondary Codes to Improve Signal Acquisition Performance in Galileo Receivers

Nagaraj C Shivaramaiah, Andrew G Dempster and Chris Rizos
University of New South Wales, Sydney, NSW 2052, Australia

BIOGRAPHY

Nagaraj C Shivaramaiah is currently a doctoral student within the GNSS receiver design group in the School of Surveying and Spatial Information Systems at the University of New South Wales, Australia. He obtained his Masters degree from the Centre for Electronics Design and Technology at Indian Institute of Science, Bangalore, India. He has been involved in GNSS related activities since late 90s. His research interests include baseband signal processing and FPGA based receiver design for GNSS.

Andrew G Dempster is the Director of Research in the School of Surveying and Spatial Information Systems at the University of New South Wales. He led the team that developed Australia's first GPS receiver in the late 80s and has been involved with satellite navigation ever since. His current research interests are GNSS receiver design, GNSS signal processing, and new location technologies.

Prof. Chris Rizos is a graduate of The University of New South Wales (UNSW), Sydney, Australia; obtaining a PhD in Satellite Geodesy in 1980. Chris is currently the Head of the School of Surveying & Spatial Information Systems at UNSW. Chris has been researching the technology and applications of GPS since 1985, and established over a decade ago the Satellite Navigation and Positioning group at UNSW, today the largest and best known academic GPS and wireless location technology R&D laboratory in Australia. Chris is the Vice President of the International Association of Geodesy (IAG), a member of the Governing Board of the International GNSS Service, and a member of the IAG's Global Geodetic Observing System Steering Committee. Chris is a Fellow of the IAG and of the Australian Institute of Navigation.

ABSTRACT

In GNSS, longer integrations are required to obtain better signal-to-noise ratio during the signal synchronization process. However the presence of secondary codes on the top of primary codes puts a constraint on the coherent integration duration for pilot channels in a similar way to the effect of data bits in data-carrying channels. In this paper we explore the problem of coherent integration over periods longer than one primary code length and the acquisition of secondary code chip position. We propose an acquisition engine architecture which can handle both these problems together.

INTRODUCTION

Longer length ranging codes are a feature of new GNSS signals as they provide good cross-correlation benefits. A method of generating longer length ranging codes is to use a slower and smaller length code called a secondary code with a faster and medium length primary code to form a longer 'tiered' code. Galileo signals employ secondary codes of varying (length N_s) combined with the primary codes (length N_p) to form the final spreading code of length N_pN_s. Galileo E1 uses a 25 chip secondary code whereas the E5 signal uses 20, 100, 4 and 100 chip secondary codes for the E5a-I, E5a-Q, E5b-I and E5b-Q respectively. The primary code period (T_p) for Galileo E1, GIOVE-A E1-B is 4 milliseconds, GIOVE-A E1-C is 8 milliseconds and all the primary codes in E5 have a period of one millisecond. The secondary codes used in Galileo signals are memory codes [1,7].

A well known method of weak signal acquisition is to integrate the correlation values for a longer period in order to achieve a good post-correlation signal-to-noise ratio, and hence allow a sufficient margin for the decision statistic to pass the acquisition threshold test. Pre-detection integration over one primary code duration is often not sufficient to acquire weak signals and the presence of a secondary code of unknown phase prevents the receiver from performing a longer integration. Knowledge of secondary code phase is required to perform longer coherent integrations.

Extended integration can be achieved by a suitable combination of coherent and non-coherent integration of the correlation values. Even though the non-coherent integration performance is inferior to that of coherent integration, it is a preferred choice in traditional receivers so as to integrate across the data-bit boundaries. Secondary codes have the same effect on the correlation values as data-bit transitions, but can be known a-priori by the receiver, unlike the data.

For a smooth transition from acquisition to tracking (using a phase-locked loop rather than a Costas loop) it is also required to acquire the secondary code chip position. Existing methods follow a two-step procedure to acquire the primary code first and then to acquire the secondary code by trying out all the possible secondary code delays. For longer integrations in weak signal environments this is a time consuming task and increases the acquisition time, especially with long secondary code lengths such as 100 chips.

Previous work related to Galileo signal acquisition considers only the primary code period for coherent integration, and then the result is integrated non-coherently for longer integration periods, e.g. [2]. A closely related work [3] which uses a multi-hypothesis technique demands larger and larger memory as the integration time is increased and also the secondary code phase has to be acquired in a separate process or one should wait till the maximum length of the secondary code is reached.

The architecture presented in this paper takes a different approach to perform longer integrations to that of non-coherently combining one primary code period correlation values. Correlation values obtained by integrating over a period T_p are coherently accumulated with succeeding values. This accumulation is performed by using knowledge of the secondary code, i.e. an output is produced for all the N_s delays of the secondary code. This coherent integration is continued for the desired duration and then the decision statistic is found by taking the maximum value among a maximum of N_s correlation values. Note that when correlations are performed with the pilot signals, the integration can be extended to any desired length as long as the receiver dynamics itself doesn't alter the code phase delay.

As the secondary codes are memory codes, the second problem of finding the secondary code chip position leads to a question. Do we need to search the entire length of the secondary code to find the chip position? This is important because the computational resource and time taken for completion of secondary code acquisition can be reduced if we can find the secondary code delay within the first few accumulations. Analysis and simulations show that out of all N_s correlation values that are accumulated in each T_p seconds, one accumulated correlation value which is a potential winner clearly distinguishes itself from the others by producing a higher and constantly increasing correlation value. This trend is seen at very early stages of the accumulation process (e.g. around chip 15 for the E5a-Q whose secondary code length is 100). This shows that we need not integrate for the entire secondary code length to identify the phase of the secondary code.

SECONDARY CODES IN GALIELO

In this section we provide a brief overview of the code structure for Galileo open service signals viz. E1 and E5, concentrating on the secondary codes. Table 1 details the code structure. Note that except for the E1-B signal component which carries only the navigation data, all the other signals have two-tiered code structure. Figure 1 illustrates the tiered code generation. Each chip of the secondary code spans one complete primary code period. In other words, the chip transition of the secondary code is aligned with the chip 'zero' of the primary code. The two codes are XORed to generate the tiered code. If N_p is the number of chips in the primary code (the length of the primary code) and N_s is the number of chips in the secondary code then the tiered code will have $N_p \cdot N_s$ chips.

The secondary codes used in Galileo are memory codes. These codes are generated to provide better correlation properties for the final pseudorandom sequences. Each code has a code identifier mnemonic as given in Table 1. For the E1-C, E5a-I and E5b-I signals, all the satellites use the single secondary code sequence $CS25_1$, $CS20_1$ and $CS4_1$ respectively. For the E5a-Q and E5b-Q signals, each satellite has different secondary code sequences (of the same length 100 bits) with the suffix distinguishing the sequences.

Signal Component	Full Tiered Code Period(ms)	Code Length (Chips)		Secondary Code Mnemonic
		Primary	Secondary	
E1-B	4	4092	1	N/A
E1-C	100	4092	25	$CS25_1$
E5a-I	20	10230	20	$CS20_1$
E5a-Q	100	10230	100	$CS100_{1\text{-}50}$
E5b-I	4	10230	4	$CS4_1$
E5b-Q	100	10230	100	$CS100_{51\text{-}100}$

Table 1 Galileo Open Service Signal Code Structure (from [1])

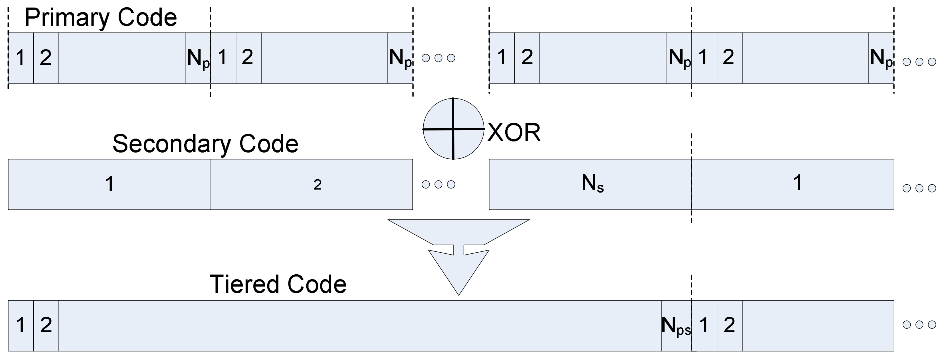

Figure 1 Tiered Code Generation

CHALLENGES IN THE PRESENCE OF SECONDARY CODES

In order to reduce the noise in the acquisition process, a typical method is to integrate the correlation samples over longer durations so as to reduce the noise bandwidth and hence the noise at the decision point. Generally integration for a single period of the primary code will not yield the best noise performance. For example, the primary codes in E5 signal have a period of 1 millisecond which offers only 30dB gain in the C/N_0, whereas each doubling this integration time increases the gain by 3dB. Hence in principle, one can keep on increasing the integration time and for the pilot signals this is often only limited by the receiver dynamics and reference clock frequency drift. In the absence of a secondary code, the integration longer than one primary code period is performed by coherently combining the successive correlation samples of one primary code length. This is possible because the primary code period is only moderately large. On the other hand, the length of the tiered code is very long to be acquired in a single step. Hence the initial task of the acquisition engine is to align the local primary code replica with the primary code boundary of the received signal. Once the primary code chip shift is found, the secondary code chip position is then acquired, thus completing the acquisition process.

The presence of secondary codes basically imposes two challenges in this process. The first challenge is the longer integration. Because the acquisition engine will not have knowledge of the secondary code chip position (and hence the chip value), the secondary code chip transition may result in loss of the coherently combined correlation value. This problem is similar to the data bit transition problem for the GPS L1 C/A signal. The simplest solution is to non-coherently combine the correlation values of one primary code length to obtain the final correlation value. However, the non-coherent combining results in a lower integration gain (mainly due to the squaring loss). A more sophisticated approach to the longer integration problem is to analyze coherent combination of all the secondary code transition hypotheses and then select the maximum among them. This approach is detailed in [3]. This results in an evolutionary tree with 2^{Nc-1} leaves (where N_c is the number of secondary code chips) whose size doubles for every additional primary code period. For example, in order to integrate for 4 primary code lengths, we need to perform and analyze correlation for 8 combinations. For 8 primary code lengths integration there will be 128 combinations; it is practically difficult to handle such high integration durations. Also note that the standard evolutionary tree approach for the longer integration does not consider the receiver's knowledge of the secondary code.

The second challenge is to acquire the chip position of the secondary code. It is required to know the secondary chip position so as to pass the information to the signal tracking stage. Without this information, the tracking process can not wipe-off the effect of secondary codes. The acquisition of the secondary code chip position is performed as a second step after the primary code acquisition. This results in additional time to be spent during the acquisition process. For example, to acquire the secondary code chip position in E5a-Q or E5b-Q signal, 100 combinations of 100 consecutive one millisecond correlation values have to be examined.

These two problems are the main motivations to explore the properties of published Galileo secondary codes which is the topic of next section.

SOME PROPERTIES OF GALILEO SECONDARY CODES

Full- sequence Auto-correlation

Figure 2 and Figure 3 show the auto-correlation plots of two of the selected secondary codes. As can be seen from the figures the codes have very good auto-correlation properties. The auto-correlation sub-peak is about 18 dB below the main peak in $CS25_1$ and about 21dB below the main peak in $CS100_1$.

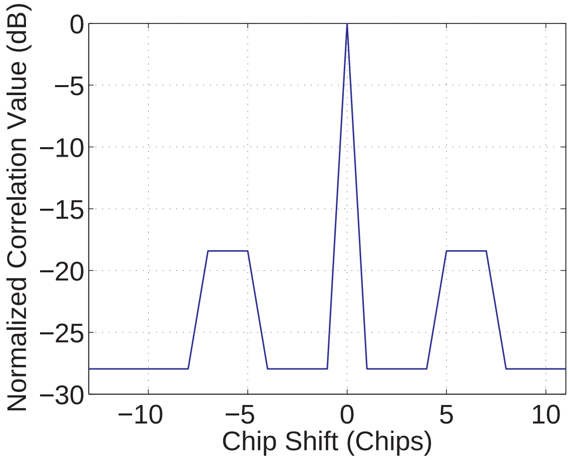

Figure 2 Autocorrelation plot of the secondary code CS25$_1$

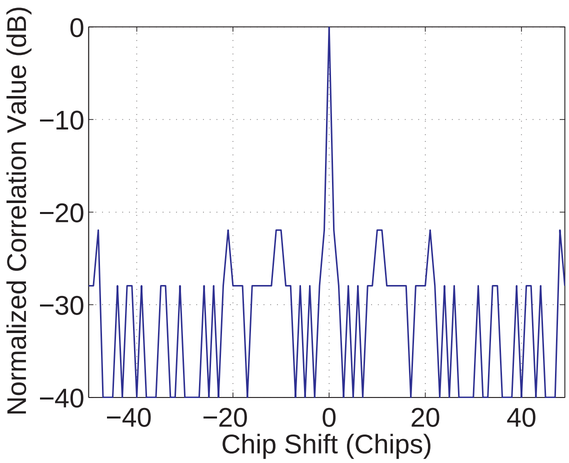

Figure 3 Autocorrelation plot of the secondary code CS100$_1$

Minimum sequence length required to identify the chip position, the Characteristic Length

For pseudorandom sequences which are *m*-sequences [5] generated by a Linear Feedback Shift Register (LFSR) of length *n*, we know that the chip position of any chip in the sequence can be uniquely identified by just looking at *n* chips (including the current). This is possible because while generating the maximal length sequence, the shift register traverses through all possible $2^n - 1$ binary combinations (i.e. except 'all zeros'). If *k* is the current chip position, the chip value c_k depends only on the previous *n* values. Mathematically,

$$c_{k+1} = f(c_{k-n-1}, c_{k-n-2}, \ldots, c_k) \qquad (1)$$

This length which is just sufficient to identify the chip shift is called the Linear span '*L*' of that code sequence [4, 6]. Note that the Berlekamp-Massey algorithm [4] can be used to reconstruct the entire sequence if we consider 2*L* chips (without the knowledge of feedback taps). However, because the memory codes are not generated through LFSR, we just make use the concept of Linear span and not the sequence reconstruction.

For the Gold codes (for example those used in GPS L1 C/A) generated with two *n*-bit shift registers, the Linear span is 2*n* chips. This implies that if we break the sequence into smaller sequences each of length equal to the Linear span, then no two smaller sequence bit patterns will be identical to each other.

We extended the above mentioned concept to the memory codes. Even though the memory codes are not generated via a LFSR, we explored their spans. We call this span the *Characteristic Length* (and use the same notation *L*) of the sequence.

Procedure for Evaluating the Characteristic Length

In order to do this, we followed these steps.

1. Let *k* be the number of contiguous 'zeros' or 'ones' (whichever is maximum) in the sequence whose length is N_s.
2. Form a matrix *M* with the partial sequences of length *k* as the rows where each row is shifted by one bit w.r.t. the previous row. Hence the size of the matrix will be $N_s \times k$.

$$M = \begin{bmatrix} c_1 & c_2 & . & . & c_k \\ c_2 & c_3 & . & . & c_{k+1} \\ . & . & . & . & . \\ . & . & . & . & . \\ c_{Ns-k} & c_{Ns-k-1} & . & . & c_{Ns} \\ . & . & . & . & . \\ c_{Ns} & c_1 & . & . & c_{k-1} \end{bmatrix} \qquad (2)$$

3. Examine the matrix for identical rows. If any two rows are identical then increment *k* and repeat the step 2 until no two rows of the matrix are identical. The uniqueness of the rows can be found by computing the linear rank correlation coefficient matrix $X = CORR(M)$ of the rows of M and examining whether any entry of X is a unity.

4. The smallest value of *k* which satisfies the condition in step 3 is the characteristic length *L* of the sequence.

The characteristic length obtained using the aforementioned procedure for different secondary code sequences of Galileo is given in Table 2. For the E5 secondary code which is different for each satellite, figure 4 shows the Characteristic lengths and figure 5 shows the histogram of the same.

Secondary Code	Characteristic Length (*L*)
CS4	3
$CS20_1$	8
$CS25_1$	7
$CS100_b$(GIOVE-A)	15
$CS100_d$(GIOVE-A)	13

Table 2 Characteristic lengths for Galileo secondary codes

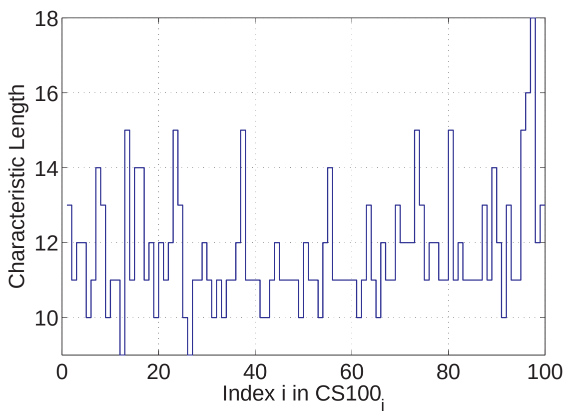

Figure 4 Characteristic lengths for the E5 secondary codes

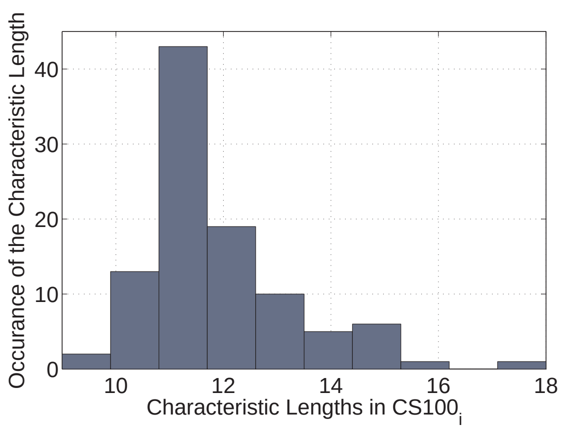

Figure 5 Histogram of the characteristic lengths of E5 secondary codes

As we can see from the table and the figures above, the characteristic lengths are much smaller than the sequence lengths and the E5 secondary codes have characteristic lengths between 9 and 18. In addition, most of the E5 sequences have a characteristic length of 11.

SYSTEM MODEL

This section describes the proposed method for longer integration and to find the secondary code chip position. We know that the coherent integration has to be extended for a duration more than one code period to achieve the required integration gain. If N_c is the number of primary code periods (or secondary code chips) used for the coherent integration, then the coherent integration duration is

$$T_c = N_c \cdot T_p \tag{3}$$

where T_p is the primary code period. Depending on the characteristic length of the code sequence under consideration, N_c may be smaller or larger than L and the problem of detecting the signal and finding the secondary code chip position has to be addressed appropriately.

Figure 6 shows the system model for the proposed approach. The primary code detector is a correlator which performs the correlation of the input signal with the local primary code replica.

Secondary code hypothesis block

The secondary code hypotheses block evaluates all the required secondary code combinations using the evolutionary tree approach described in [3]. This block evaluates B_i branches at a time.

If N_c << L, the value of B will be 2^{Nc-1} to start with. Hence

$$B_i = \begin{cases} 2^{N_c-1}, i = 0 \\ B_{i-1} - E_i, i \neq 0 \end{cases} \tag{4}$$

where E_i is the number of branches eliminated in the i^{th} iteration.

If N_c >>L or $N_c \approx L$, then the value of B in each iteration will depend on the strength of the received signal. This means that as we extend the integration time, the branch elimination logic can better decide on the branches to be eliminated.

Branch elimination logic

The branch elimination logic examines all the hypotheses output by the secondary code hypothesis block. The criterion for any branch elimination is the lower correlation value relative to other branches. Let $S = (\hat{s}_1, \hat{s}_2, \ldots \hat{s}_{Bi})^T$ be the vector containing the entire secondary code hypotheses, where $\hat{s}_i$ is the i^{th} secondary code hypothesis. We form a vector D containing the difference w.r.t. the maximum. Thus we compute D as

$$D = \max(S) - S \tag{5}$$

We eliminate those branches which are above a predefined threshold η and hence we obtain E_i, the number of branches eliminated.

Decision to end the iteration

When $N_c << L$ (which is the case when the signal strength is moderately high) we would have detected the signal before we determine the secondary code chip position. In this case we need to continue the iteration to determine the secondary code chip position and hence the emphasis is on the crossing point of the characteristic length. Thus the iteration ends when the total number of primary code periods used in all the iterations is greater than L. At each stage, the secondary code is advanced by K number of chips which is same as N_c.

When $N_c >> L$ or $N_c \approx L$, (which is the case when the signal strength is less) we would have to integrate long enough so that the decision statistic

$$\Lambda = \max\{\lambda(s_i)\},\ i = (0, B_i - 1) \tag{6}$$

is greater than the threshold ξ, to detect the signal. Here $\lambda(s_i)$ is the detector output of i^{th} secondary code hypothesis [2]. Hence in this case the emphasis is on the signal detection and the hypothesis which causes the decision statistic to cross the threshold is used to determine the secondary code chip position. At each stage, the secondary code is advanced by K number of chips that is sufficient to eliminate some of the branches.

Secondary code chip position retrieval

The secondary code hypothesis that we obtain is a sub-sequence within the complete secondary code. Thus retrieving the chip position is performed by searching for this sub-sequence in the larger sequence and determining the index of the shift.

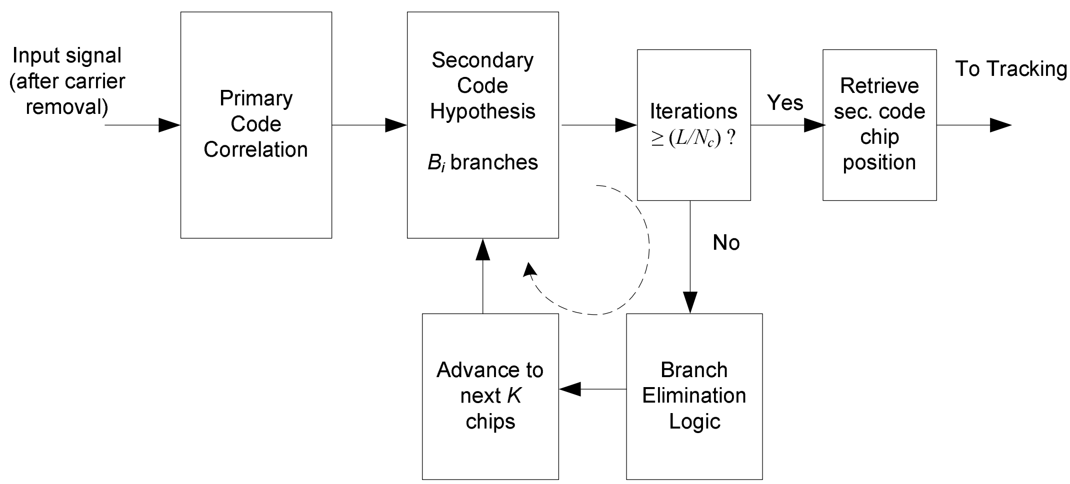

Figure 6(a) Proposed system model when $N_c << L$

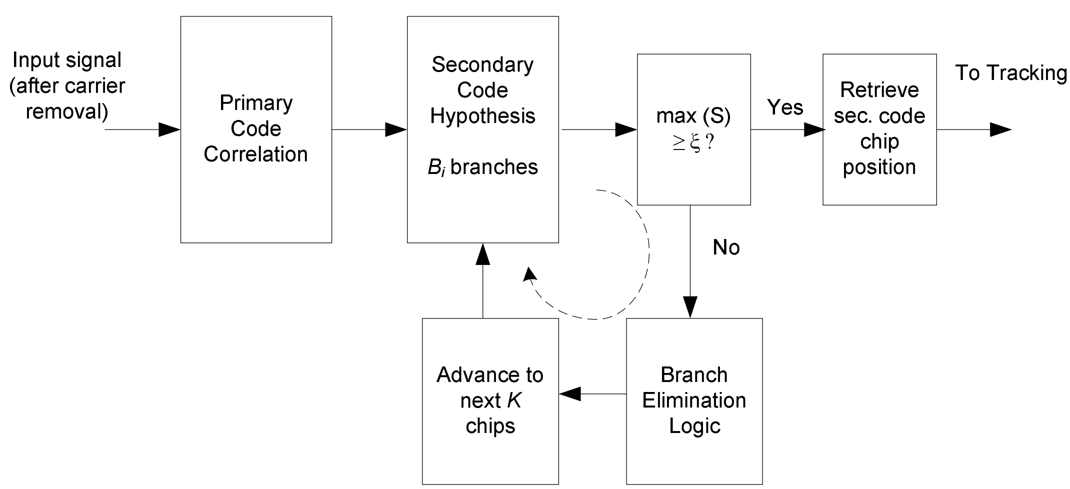

Figure 6(b) Proposed system model when $N_c >> L$ or $N_c \approx L$

RESULTS

In order to evaluate the performance of the proposed method, we used 150ms of real data collected from Septentrio GeNeRX1 receiver for the E5 signal from the GIOVE-A satellite. Determining the secondary code chip position is severe in the case of E5 since the primary code period is only one millisecond and the secondary code 100 chips for the pilot signals.

The procedure followed to acquire the E5 signal is as follows.

1. Acquire E5aQ pilot signal whilst finding the secondary code chip position
2. Use this information of the secondary code chip position and acquire E5 signal

Figure 7 shows the trend in correlation values for increasing integration time. To show the applicability of the proposed method we used all the 100 hypotheses. Observe that there is only one potential winner, which can be clearly distinguished from other hypotheses as the integration time increases. The deviation point of other sequences compared to the potential winner depends on

i. the hamming distance of the potential winner with respect to the other sub sequences (in the *X* matrix) and
ii. the position of the chip differences that result in this hamming distance.

As an example consider a sequence with characteristic length of 15. If the minimum hamming distance of the potential winner w.r.t. other sub-sequences is 4 (say) and the chip differences appear after 11, then the closest contender grows with the winner before deviating at an integration time interval of 11 ms. For the same case of hamming distance of 4, if there are some bit differences early on in the sequence then the closest contender will grow but in a parallel track below the potential winner eventually deviating at 11ms. Figure 7 used GIOVEA $CS100_b$ code (whose characteristic length is 15) and in the received signal, the winning sub-sequence was located starting 46^{th} chip of the secondary code sequence. Without loss of generality we conclude that in all cases this deviation point will occur ahead of the characteristic length.

Figure 8 shows the correlation plots for different integration durations. Note that the correlation values of different hypotheses are close to each other when the integration time is less than the characteristic length (because the primary code period is 1 millisecond for E5, the terms 'integration time' and 'number of primary code periods can be used interchangeably). As soon as the integration time reaches the characteristic length, a clear peak pops out.

The secondary code chip position obtained in this case was 46 which corresponds to the sub-sequence [1 0 0 0 1 0 1 1 0 0 0 1 1 1 0]. Using this information we acquired the E5 signal with 1 millisecond and 4 millisecond integrations. The plots (and the zoom versions around the peak) are shown in figure 9 and 10.

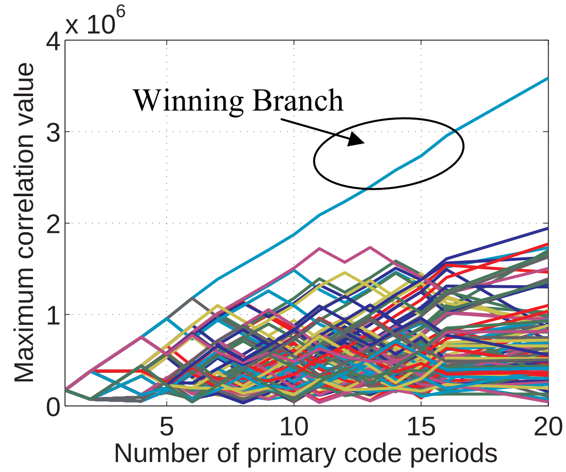

Figure 7 Correlation value trend for increasing number of primary code period integrations (different colors show all the 100 hypotheses)

CONCLUDING REMARKS

We conclude that the acquisition architecture presented in this paper is unique in the sense of achieving longer integration by exerting secondary codes and also acquiring the secondary code chip position as a by-product of the acquisition process. It is shown that the secondary code chip shift can be uniquely identified by shorter length sequences than the code itself. Most of the E5 secondary codes of length 100 can be identified by shorter sequences of length around 15.

ACKNOWLEDGMENTS

The authors would like to acknowledge that this research work has been carried out under the Australian Research Council (ARC) project DP0556848.

REFERENCES

[1] Galileo Open Service Signal In Space Interface Control Document (OS SIS ICD), Draft 1, February 2008

[2] De Wilde W, et.al. "Fast signal acquisition technology for new GPS/Galileo receivers", IEEE/ION PLANS 2006, April 25-27, 2006, pp. 1074-1079

[3] C Palestini, et.al. "Galileo Primary Code Acquisition Based on Multi-hypothesis Secondary Code Ambiguity Elimination", ION-GNSS07, 25-28 Sep 2007, Fort Worth, TX, pp. 2459-2465

[4] Giovanni Cherubini, Nevio Benvenuto, Algorithms for Communications Systems and their Applications, Wiley Publications, October 18, 2002

[5] Dilip V. Sarwate, Michael B. Pursley, "Cross-correlation Properties of Pseudorandom and Related Sequences", *Proceedings of the IEEE*, vol 68, no 5, May 1980.

[6] Agnes Chan, Richard Games, "On the Linear Span of Binary Sequences Obtained from Q-Ary M Sequences, Q Odd", *IEEE Transactions on Information Theory*, vol 36, no 3, May 1990.

[7] GIOVE-A Navigation Signal-In-Space Interface Control Document, Issue 1, Revision 0, March 2007.

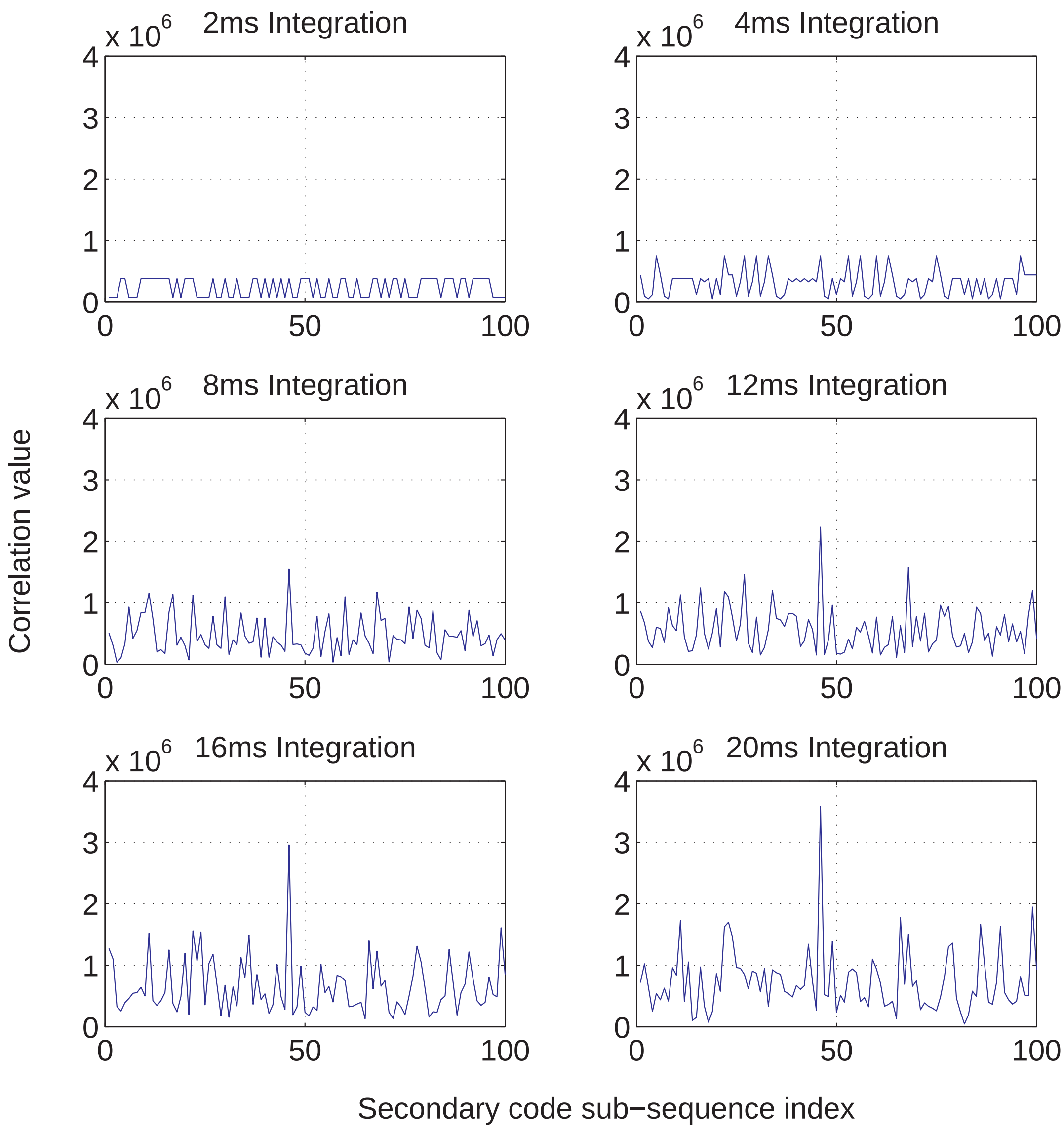

Figure 8 Correlation values for all the secondary code hypotheses (sub-sequence indices)

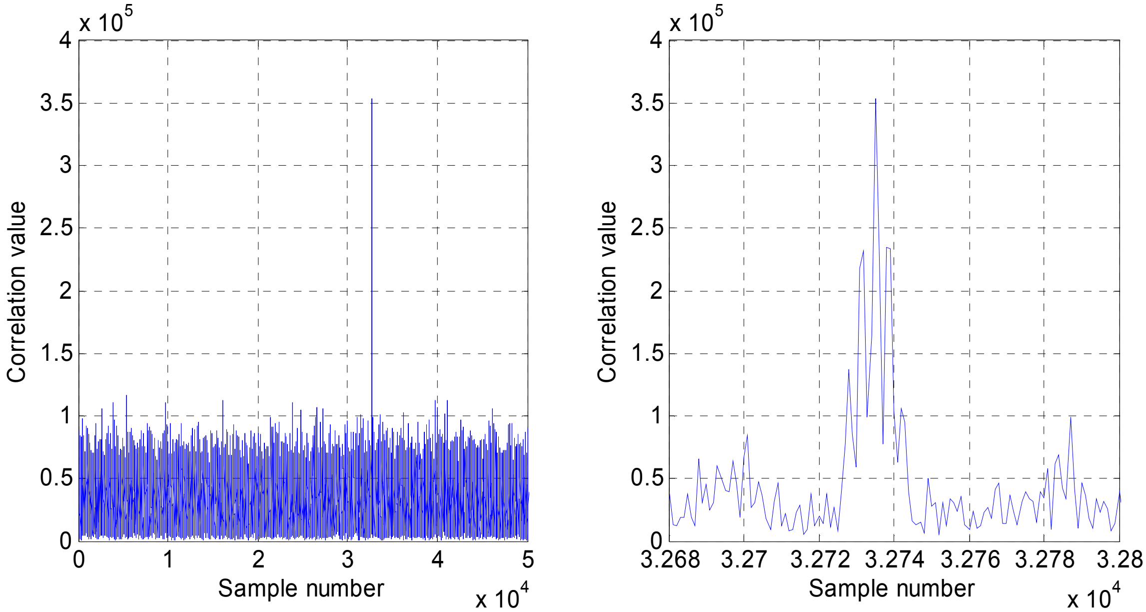

Figure 9 Absolute correlation value of the E5 signal (right picture is the zoom version around the peak); 1ms integration using the secondary code chip position detection algorithm

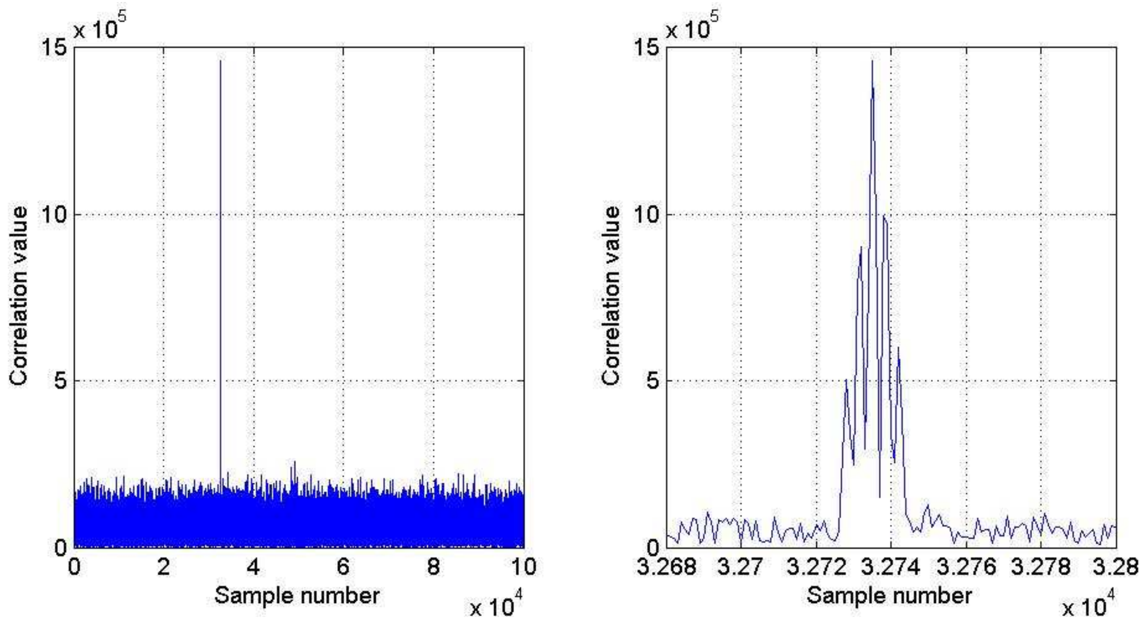

Figure 10 Absolute correlation value of the E5 signal (right picture is the zoom version around the peak); 4ms integration using the secondary code chip position detection algorithm; data collected for GIOVE-A satellite on 5th Feb 2008