

## Use of comb filters in GPS L1 receivers

**Author:**

Dempster, Andrew

**Publication details:**

GPS Solutions

v. 12

Chapter No. 3

pp. 179-185

1080-5370 (ISSN)

**Publication Date:**

2008

**Publisher DOI:**

<http://dx.doi.org/10.1007/s10291-007-0079-3>

**License:**

<https://creativecommons.org/licenses/by-nc-nd/3.0/au/>

Link to license to see what you are allowed to do with this resource.

Downloaded from <http://hdl.handle.net/1959.4/44318> in <https://unsworks.unsw.edu.au> on 2024-03-28

# Use of Comb Filters in GPS L1 Receivers

ANDREW G DEMPSTER (✉)

*School of Surveying and Spatial Information Systems,*

*University of New South Wales,*

*Sydney, 2052, New South Wales,*

*Australia*

a.dempster@unsw.edu.au

+61 2 93856890 (ph)

+61 2 93137493 (fax)

## Abstract

Comb filters can be designed by replacing single delays in digital filters by multiple delays. These filters are “periodic” in the sense that their frequency response repeats itself at fixed intervals. They are thus well suited to a signal such as GPS L1 which has a spectrum made up of lines separated by the code repetition frequency of 1 kHz. In this paper we examine how periodic filters can be used in a conventional GPS L1 receiver. We find that comb filters assist with acquisition of weak signals, and can improve code tracking performance.

**Keywords:** GPS Receivers; Digital Filters; Comb filters; Signal Processing

## Introduction

The GPS L1 signal consists of a 50 bits/sec data stream “spread” by a pseudorandom code at 1.023Mchip/sec. This code is 1023 chips long and repeats at 1ms intervals. Because of this repetition, the spectrum of the spreading code has a “sinc-shaped” envelope that scales lines 1kHz apart. This line spectrum, centred at the carrier frequency of 1.57542GHz and with each line convolved with the 50 bit/sec data, is the spectrum of a transmitted GPS L1 signal.

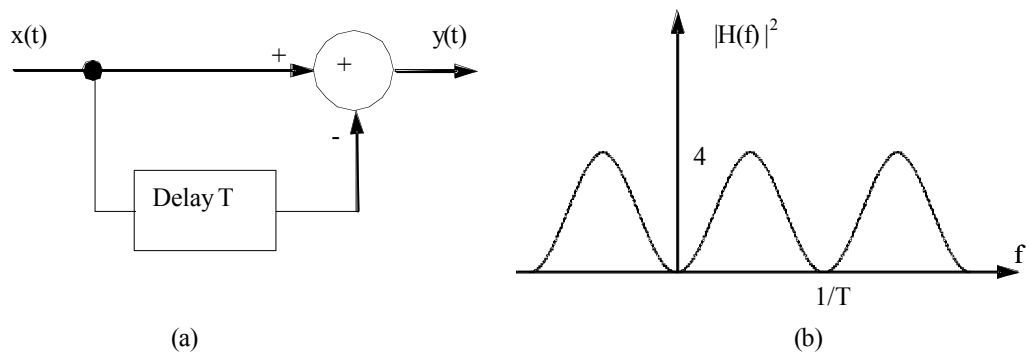
If it was possible to design a filter that passed these lines, while attenuating the “noise” in between the lines, then the SNR of the received signal would be improved, allowing, for instance, weaker signals to be detected and tracked. Because such a filter would have passbands evenly spaced at 1kHz intervals, this type of filter can be described as “periodic”. A comb filter meets these criteria.

Recently, comb filters were examined in an acquisition scheme for GPS L1 (Shanmugam et al 2006). In that paper, the comb filter, which was described simply as a “pre-filter”, was used to enhance the behaviour of differential detection schemes used by the same authors (Shanmugam et al 2005). The use of periodic filters to enhance differential detection was first introduced in (Coenen and Van Nee, 1992). Here we examine the application of comb filters to conventional correlator designs.

# Comb Filters

## Analogue Comb Filters

The comb filter has found applications for over six decades. Once also called the North filter (North 1943), it has seen application for instance in television (Arndt et al 1973) and radar (George and Zamanakos 1951). In those latter two references the two key types of comb filtering are addressed: the attenuation of periodic interference and the enhancement of wanted periodic signals, respectively. In the former, the frequencies  $mf_r$  are attenuated for integer  $m$  and  $f_r$  being the frequency separation between stopbands. For the latter type of filter, the comb filter has passbands at  $mf_r$ . Both feedback and feed-forward techniques were used.



**Figure 1 a) Comb filter structure, b) Power spectrum of the comb filter**

The simplest comb filter structure is the feed-forward structure in Figure 1a, where a delayed version of the input is subtracted from the input. This has impulse response

$$h(t) = \delta(t) - \delta(t - T) \quad (1)$$

which in turn has frequency response

$$H(f) = 1 - e^{-j2\pi fT} \quad (2)$$

and power spectrum

$$|H(f)|^2 = 4 \sin^2 \pi fT \quad (3)$$

as in Figure 1b. Clearly, the spectrum is periodic. Such a filter nulls out dc and all multiples of the frequency  $1/T$ , which makes it a good filter for removing any interfering signal of period  $1/T$ . A second type of filter is produced if the subtraction in Figure 1a is replaced by an addition, when the power spectrum is

$$|H(f)|^2 = 4 \cos^2 \pi f T \quad (4)$$

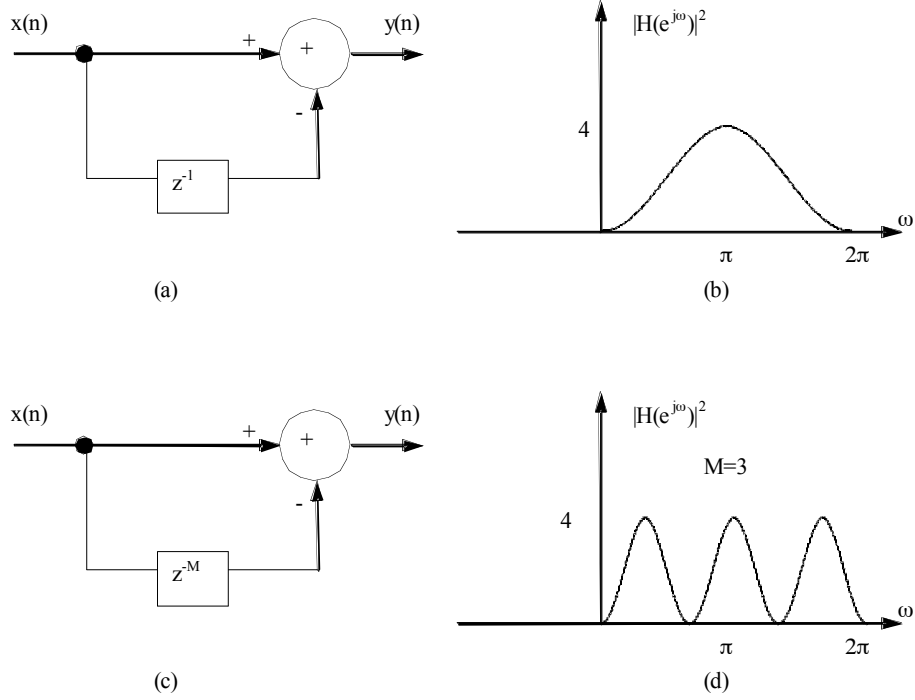
which instead has peaks at dc and multiples of  $1/T$ , thus enhancing periodic signals of period  $1/T$ . It is this second type of “enhancement” filter that is more of interest for GPS, as the signal’s repetitive code creates a line spectrum with 1kHz separations.

If an analogue comb filter such as this was to be implemented in a GPS receiver, obviously this must occur prior to analogue-to-digital conversion. This is not practical for a number of reasons. First, the satellite signals will each be experiencing a different Doppler frequency ranging over several kHz. Therefore there is not a single “frequency” that can be identified as a “base” frequency for a comb filter that would allow all satellite signals to pass. Second, the two simple designs above only have peaks at either  $(m + \frac{1}{2})$  kHz or  $m$  kHz (where  $m$  is an integer), and Doppler shift ensures that signal frequencies will not match these for most of the time. Third, the delay  $T$  must be accurately implemented, as it sets the frequency separation of the passbands. The absolute offset of the passband centre from its design position increases proportionally to  $m$ , the number of the harmonics. In most comb filter applications, this is not critical, as higher harmonics of a signal usually decay rapidly so for higher values of  $m/T$ , the signal that leaks through (in the interference removal case) will have been weak. In our application, though, we wish for the whole GPS bandwidth (2MHz) to have good alignment between the spectral lines and the filter passbands. Designing an analogue delay of that accuracy requires expensive components. For these reasons, we should consider performing the comb filtering in the digital domain.

## Digital Comb Filters

A digital comb filter can be designed by manipulating a simple digital filter design. For example, the high-pass design in Figure 2 is turned into a comb filter for removing interference with harmonics by replacing the delay by  $M$  delays.

The spectrum for  $M=3$  is shown in Figure 2d. This process, of replacing a prototype filter having transform  $H(z)$  with  $H(z^M)$ , can be used to design a comb filter with arbitrary transfer function depending on the prototype selected (see for example Lim, 1986).



**Figure 2 A simply high-pass filter (a) and its spectrum (b) can be turned in to a comb filter (c) by using  $M$  delays instead on one giving a periodic spectrum, e.g. for  $M=3$  (d)**

Turning to GPS applications, the comb filter used in (Coenen and Van Nee, 1992) was a simple accumulator, as in Figure 3a whereas in (Shanmugam et al 2006), the accumulators were longer using more delays in the prototype filter such as in Figure 3c where the accumulation occurs for  $N=2$  delays. In the case of Figure 3c, the prototype filter has  $z$ -transform  $H(z) = 1 + z^{-1} + z^{-2}$ . Longer accumulations would use as the prototype

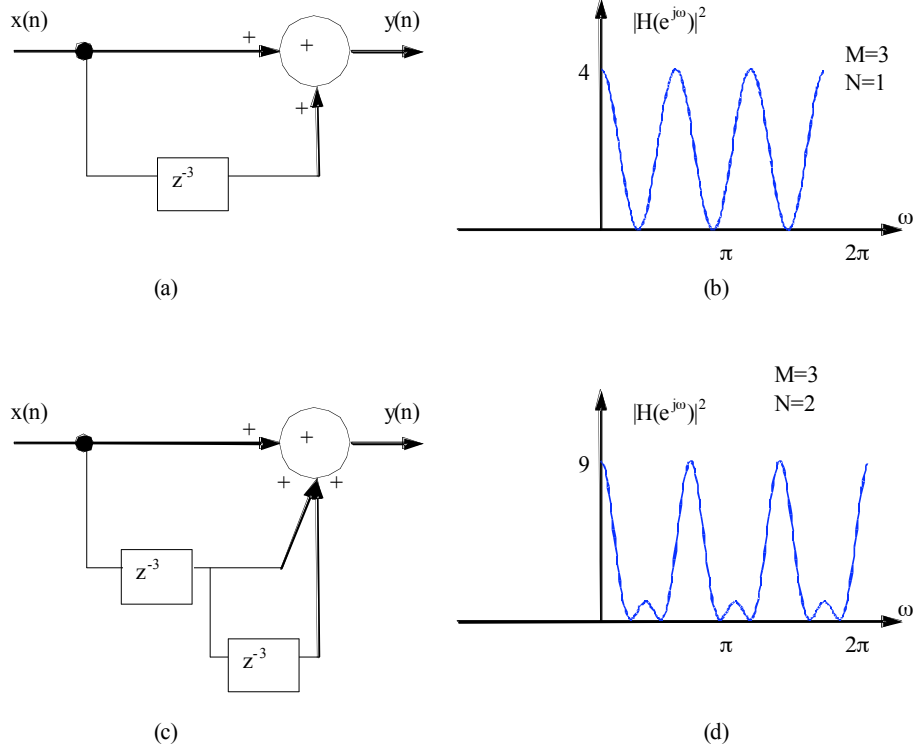
$$H(z) = \sum_{k=0}^N z^{-k} \quad (5)$$

giving a comb filter with  $z$ -transform

$$H(z) = \sum_{k=0}^N z^{-kM} \quad (6)$$

and thus response

$$H(z) = \sum_{k=0}^N e^{-jkM\omega} \quad (7)$$



**Figure 3 A comb filter (a) using a simple single-delay accumulator prototype filter and  $M=3$ , and its spectrum (b). A  $N=2$  delay accumulator with  $M=3$  (c) and its spectrum (d).**

The prototype filter used to produce the comb filter need not be a finite impulse response (FIR) filter, but can incorporate feedback, using an infinite impulse response (IIR) filter, which can produce a more tightly constrained frequency response for the same order of filter, but must be designed such that it is stable. It will also not have linear phase.

If a digital comb filter is used in a GPS receiver directly at IF, it experiences several of the problems listed above for the analogue comb filter. Again, the sampled IF data has signals at several Doppler frequencies so a single filter cannot service all satellites at IF. Also, the filters discussed are centred at dc which will not necessarily line up with the satellite Doppler frequencies. Finally, as for the analogue case, if the peaks of the comb filter are to line up with the GPS spectral lines, the  $M$  delays must equate to 1ms.

These three problems are now discussed in order, with reference to Figure 4. The first problem simply implies that a single filter cannot be used at the IF frequency to deal with all the received satellite signals (position A in Figure 4), i.e. each satellite channel must have its own comb filter.

The second problem implies that, if an individual signal is to be dealt with at the IF (position B in Figure 4), the comb filter will need to have its passbands aligned to the carrier at that point. The frequency offset of the carrier from 0Hz,  $\omega_0$ , is due to the Doppler effect, clock drift and the effects of downconversion. Its variation is often simply referred to as “Doppler”. For the passbands of the comb filter to match the signal with offset  $\omega_0$ , the filter needs to have the response (single-sided) of:

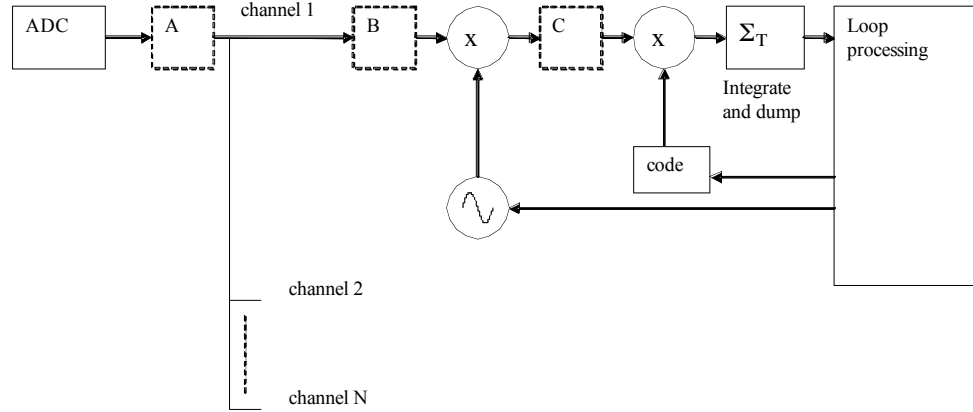
$$H(z) = e^{j\omega_0} \sum_{k=0}^N e^{-jkM\omega} \quad (8)$$

which requires complex coefficients, or the more usual double-sided response (that passes both positive and negative frequencies):

$$H(z) = \cos \omega_0 \sum_{k=0}^N e^{-jkM\omega} \quad (9)$$

Both of these filter responses would need much more complex structures than the simple comb filter, and both require a good estimate of  $\omega_0$ , which in most cases exists in a receiver anyway because it is needed for the carrier tracking loop. In other words, if the comb filter is located at position C in Figure 4, i.e. after the carrier has been wiped, then it will still have a simple design and will exploit the existing estimate of carrier frequency. We therefore assume henceforth that the comb filter is at location C.





**Figure 4. Candidate locations for the comb filter. A: at the IF, for all satellites, B: at the IF for each satellite separately, and C: at dc after carrier wipe for each satellite separately. Location C is that examined in this paper.**

The third problem is more interesting. If the M-sample delay equals 1ms, the signal must be sampled at an integer multiple (M) of 1kHz. In general, this is the case with most existing front end chip sets so this in itself tends not to be a problem. The case of a non-integer relationship between sampling rate and delay for comb filters has been examined (Pei and Tseng, 1998), using standard interpolation techniques, but again this introduces a high level of complexity, replacing each delay with a relatively complex structure, so for our purposes, the integer relationship is assumed.

It may appear that the “exact” 1ms delay could cause a problem in this application. In the historic interference elimination application of comb filters, the requirement has been to eliminate dc, the fundamental and some harmonics of a periodic signal. Very few (say 10) harmonics need to be removed before they dwindle in power. The same attenuation of higher harmonics is also true of the signals that can be enhanced by comb filters. However, in this application, a very large number of spectral lines must be passed by the periodic passbands. In the first lobe of the L1 signal’s bandwidth, there are  $1.023\text{MHz}/1\text{kHz} = 1023$  lines each of which requires a passband that must be accurately placed at 1kHz intervals so the requirement for an “exact” 1ms delay is much tighter than in other applications. A quick calculation can confirm that this is not necessarily a concern. The sources of a mismatch in the signal code epoch of 1ms and the sampling rate divided by M are: i) Doppler shift will affect the length of the

received code epoch and ii) the receiver crystal may have drifted. Doppler shift is of the order of 5kHz, or 3ppm, which should dominate the drift in even the cheapest crystal, so we can ignore sampling clock drift. A narrowband front end will pass 2MHz, or  $\pm 1$ MHz at baseband. This means that the misalignment between the received code and the highest passband required will be of the order of 3Hz if the Doppler difference is 3ppm. The passbands are nowhere near this narrow (the 3dB bandwidth of the filter in Figure 3a is  $\pm 250$ Hz for 1kHz passband spacing – for 10ms integrations, or  $N=9$ , which we examine in more detail later, the bandwidth is  $\pm 45$ Hz ), so all the lines will be readily passed. If a wideband 20MHz front end is used, this effect is still only 30Hz, which passes even for  $N=9$  with little attenuation.

## Characterising the Effect of the Comb Filter

In the frequency domain, the comb filter can be thought of as a noise-reduction filter, attenuating noise in between the GPS spectral lines. In the time domain, it can be thought of as adding together the “same” point in consecutive code epochs, i.e. at the same delay through a given chip within the code epoch of 1ms. The output of the filter is:

$$y(n) = \sum_{k=0}^N x(n - kM) \quad (10)$$

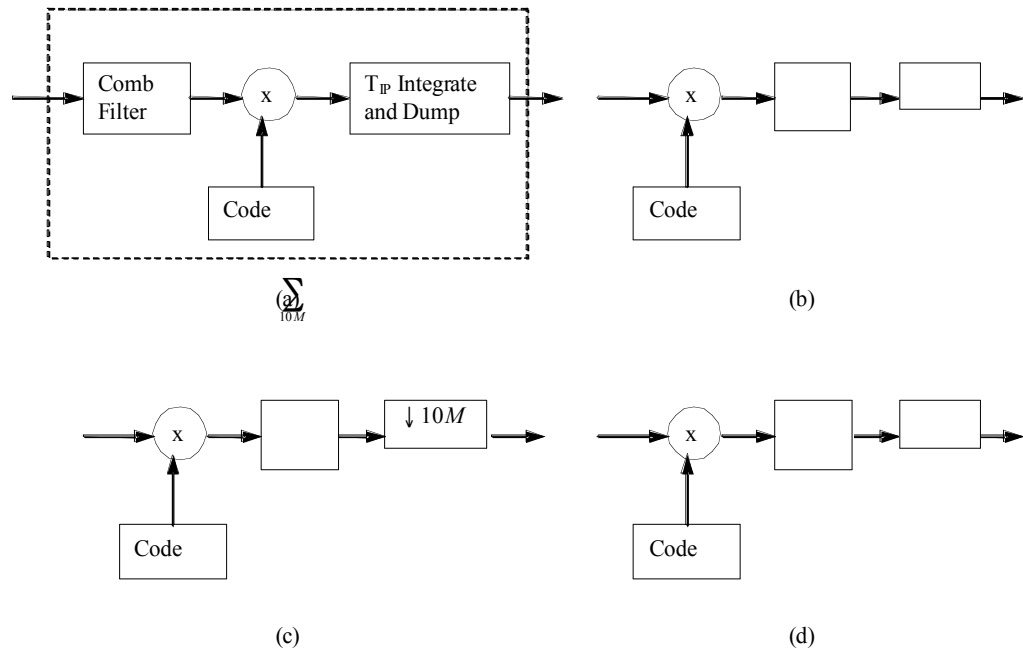
If the integrate and dump filter following the code wipe in Figure 4 has an integration period of  $T_{IP}=1$ ms, i.e.  $M$  samples, its output is

$$z(n) = \sum_{l=0}^{M-1} y(n-l) = \sum_{l=0}^{M-1} \sum_{k=0}^N x(n - kM - l) = \sum_{p=0}^{(N+1)M-1} x(n-p) \quad (11)$$

In other words, every 1ms, the integrate and dump filter produces an output that is the coherent integration of the previous  $(N+1)$  ms of samples. If the integrate and dump filter has coherent integration time  $T_{IP}=q$  ms, then its output is effectively the sum of  $q$  overlapping  $(N+1)$  ms periods.

*Example 1.* This example is used several times later in the paper for different comparisons so it is highlighted now. The case we examine is if there were  $N=9$  delay structures  $z^{-M}$  in Figure 3c), and the integrate and dump filter had  $T_{IP}=1$  ms. (Typically this is later compared with conventional architectures with  $N=0$ , and either  $T_{IP}=1$  ms or  $T_{IP}=10$  ms)

Following the above discussion, it can be seen that for Example 1, each dump from the integrate and dump filter, made at 1ms intervals, would in fact be a coherent integration of the previous 10ms of data. An attempt has been made to show this as equivalent circuits in Figure 5. Note however, that the equivalence holds only for the Figure 5d when the code is not slipped from one 1ms epoch to the next. In fact, the real comb filter circuit is more versatile than Figure 5d because the code can change from epoch to epoch and still gain the full 10ms worth of integration.



**Figure 5** Equivalent circuits for part of Figure 4 for the three situations discussed in Example 1.  $M$  is the number of samples in 1ms, i.e.  $f_s/1\text{kHz}$ . The integrate and dump function is replaced by an integrator of a number of samples (here  $M$  or  $10M$ ) and a downsampler (by factors  $M$  or  $10M$ ). (a) The part of the circuit to be replaced, (b) equivalent circuit for  $T_{IP} = 1\text{ms}$  and no comb filter, (c) for  $T_{IP} = 10\text{ms}$  and no comb filter, (d) for comb filter with  $N=9$  and for  $T_{IP} = 1\text{ms}$ .

# Acquisition

## Sequential Hardware Acquisition

From an acquisition point of view, the preceding argument seems to indicate an apparent advantage: the input samples are “integrated” in the comb filter prior to being wiped of code, so to run different code delay trials, the code delay can be changed at the rate at which the integrate and dump filter operates while still producing outputs that represent longer intervals of integration. We will assume for the remainder of this discussion the values from the Example 1: integrate and dump period  $T_{IP} = 1\text{ms}$  and  $N = 9$  leading to an effective 10ms integration. We compare with the  $T_{IP} = 1\text{ms}$  and  $T_{IP} = 10\text{ms}$  cases without a comb filter.

For sequential acquisition, to search the whole range of 1023 chips at half-chip intervals, a total of 2046 trials, using  $N_C$  correlators would take  $2046/N_C + N$  ms. If a comb filter is not used, and  $T_{IP} = 10\text{ms}$ , the search time would be  $20460/N_C$  ms, about 10 times longer if  $N_C$  is small. However, there is one key difference between using the comb filter method and the usual search with  $T_{IP} = 10\text{ms}$ . In the latter case, in two subsequent trials one is guaranteed not to experience a data transition. In the comb filter case, a data transition appears in  $N$  consecutive trials. This need not present a problem if the following solution is implemented:

*Algorithm 1:* The aim of this algorithm is to ensure that two correlator delay trials, separated by half a chip, do not both integrate over a potential data bit boundary. This can be achieved if those trials apply to data 10ms apart. This in turn can be achieved by ordering the trials through an individual correlator. If a truly sequential set of trials, stepping half a chip per trial are numbered 1, 2, 3, ..., then what we are trying to achieve is that 1 and 2, for instance do not both “see” a data bit boundary. Given that trials are occurring at a 1ms rate, and apply to the previous 10ms, if one trial “sees” that boundary the next will also (in 9 of 10 cases, the other being when the boundary was 9ms ago in the first trial). The trials can be reordered as follows: 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20 (then 21, 23 etc). In other words, the correlator steps one chip each trial, then “comes back” after 10ms and fills in the gaps. This gives identical

acquisition performance (sensitivity) to the sequential search with  $T_{IP} = 10\text{ms}$ , but at about 10 times the speed. However, it makes control of the correlator stepping more complex in the code search. If a more sophisticated algorithm such as the Tong detector (Kaplan and Hegarty, 2006) is used for the non-comb filter case, this factor will be less than 10.

Although the use of the comb filter allows the code search time to be reduced by a factor of 10 in Example 1 (compared with  $T_{IP} = 10\text{ms}$ ), it does not have the same effect in the frequency search. The  $N=9$  comb filter has frequency response identical to the  $T_{IP} = 10\text{ms}$  integrate-and-dump filter so the search in the frequency domain is still 10 times longer than for the  $T_{IP} = 1\text{ms}$  case. So, in round figures, the use of a  $N=9$  comb filter with  $T_{IP} = 1\text{ms}$  is ten times as fast as a conventional  $T_{IP} = 10\text{ms}$  search for the same sensitivity but 10 times slower than the less sensitive conventional  $T_{IP} = 1\text{ms}$  search.

There is an argument in (Shanmugam et al 2006) that states that the frequency search is reduced from the whole Doppler range to simply a 1kHz range because at some point in that range, the comb filter must pass the signal. This is not true in general for conventional receivers, because the signal will simply be downconverted to a multiple of 1kHz, not necessarily baseband. For the differential techniques discussed in (Shanmugam et al 2006), however, there is an immunity to Doppler offset so in that case the contention is true. The price those techniques pay for this immunity is a loss of sensitivity.

## **Parallel Hardware Acquisition (Search Engine)**

In the case of acquisition using a large parallel bank of correlators, or search engine, there seems little to be gained from using a comb filter prior to the bank of correlators. The benefits that accrued in the previous section were largely due to the serial nature of the search – there is virtually no difference between using a comb filter that has  $N = 9$ , followed by a  $T_{IP} = 1\text{ms}$  integrate and dump and a single  $T_{IP} = 10\text{ms}$  integrate and dump. Both have the performance of a  $T_{IP} = 10\text{ms}$  search, and both take 10ms if correlations are performed in parallel. If a data bit transition occurs, both methods require a wait of 10ms before a period not containing a transition can be guaranteed.

## Software Acquisition

In software acquisition, the whole correlation function is generated by taking the inverse discrete Fourier transform (IDFT) of the product of the signal discrete Fourier transform (DFT) and the conjugate of a stored DFT. That stored DFT could be either simply that of the required code, or the product of that code with one of the required Doppler frequencies. We consider these cases separately.

Where the full Doppler range is incorporated in the stored DFT, a single block of data is recorded and all of the stored DFTs are used in the search – the search time is thus the time to record the data plus all of that processing time. A hardware comb filter cannot be used in this scenario, because (see the argument using Figure 4) it is not aligned with the Doppler frequency (or frequencies, as all the satellites will be acquired from the same data). Running a comb filter in software would appear clumsy – a different filter is required for each potential Doppler frequency. A potential solution is to run a “block” of comb filters on the block of input data producing one new block of data for each Doppler frequency, and each satellite that is searched at that frequency would use the corresponding comb-filtered data. The advantage of doing this is that it reduces the size of the data block to 1ms from whatever length it was (10ms in Example 1), reducing both the required data storage and the amount of time spent of DFTs. However, the downfall of this method is that although “circular convolution” is not affected by the 1ms repetition rate of the data, it does affect the cases where the Doppler candidate is not an integer multiple of 1kHz (i.e. most cases). Therefore in this type of software acquisition, comb filters offer no advantage.

The second case is where Doppler is stripped before sampling and the stored DFT is that of the code only. We continue to examine Example 1 and compare with integrate and dumps of  $T_{IP} = 1\text{ms}$  and  $T_{IP} = 10\text{ms}$ , so the following can be said. For the comb filter case, an DFT-based correlation is applied after a trial Doppler downconversion. For speed, the DFT will be implemented by the FFT algorithm. That FFT needs only to cover 1ms of sampled data rather than 10ms for the  $T_{IP} = 10\text{ms}$  case. Because of the  $n \log(n)$  relationship between FFT length and its processing time, this gives a saving in processing time (or effort) of

$$\frac{T_{T_{IP}=10ms}}{T_{T_{IP}=1ms}} = 10^{\frac{\log_2(f_{smp}) - 6.6}{\log_2(f_{smp}) - 10.0}} \quad (12)$$

where  $f_{smp}$  is the ADC sampling frequency. So for a sampling frequency of 5MHz, the saving is about 12.7. Comparing with the  $T_{IP} = 1ms$  case, the FFT effort is the same but the frequency search as for sequential searching is 10 times as long, but performance (sensitivity) is improved.

## Tracking

### Code Loop

It is easiest to consider the comb filter effects on the code loop by thinking in the frequency domain. If the structure of Figure 4, with comb filter at position C, is used, the code-locked loop can be considered to be accepting a signal that has been passed through a noise-reduction filter. Once again, using Example 1, the SNR of that input is improved by 10dB, yet the code loop has the same update rate and dynamics as a  $T_{IP} = 1ms$  filter. In other words, the loop will perform better using a comb filter.

### Carrier Loop

In the case of the carrier loop, there is less of an advantage. Using Example 1 will mean that the loop filter has the same bandwidth as a conventional loop with  $T_{IP} = 10ms$ . The only “benefit” is that the loop can be updated every 1ms. However, given that any update still takes 10ms to flow completely through the loop filter, the carrier loop does not benefit greatly from using a comb filter.

## Conclusion

Comb filters have been described and shown to be useful in GPS L1 receivers. If a comb filter is to be used, it should appear after the carrier wiping and before the code wiping, in both acquisition and tracking applications. For the price of some simple extra hardware, a comb filter can accelerate sequential acquisition for longer integration periods, at the further cost of a more complicated correlator trial delay sequencer. It can also help in software acquisition, although typically it

will not. Code tracking can be enhanced but carrier tracking does not benefit significantly.

## References

Arndt G, Stuber F, Panneton R (1973), "Video-Signal Improvement Using Comb Filtering Techniques", IEEE Trans Communications, 21(4), April 1973, pp331-336

Coenen A J R M and Van Nee D J R (1992), "Novel Fast GPS/Glonass Code-Acquisition Technique Using Low Update Rate FFT", Electronics Letters, 28 (9), 23 April 1992, pp863-865

George S F and Zamanakos (1951), "Comb Filters for Pulsed Radar Use", Proc IRE, no 12, July 1951, pp1159-1165

Kaplan E D and Hegarty C (2006), "Understanding GPS", Artech House, 2<sup>nd</sup> ed, 2006

Lim Y (1986), "Frequency-response masking approach for the synthesis of sharp linear phase digital filters", IEEE Trans Circuits and Systems, 33(4), Apr 1986, pp357 - 364

North D O (1943), "Analysis of Factors which Determine Signal-Noise Discrimination in Pulsed Carrier Systems", RCA report PTR-6C, Princeton NJ, June 1943

Pei S-C and Tseng C-C (1998), "A Comb Filter Design Using Fractional-Sample Delay", IEEE Trans Circuits and Systems, 45(6), June 1998, pp649-653

Shanmugam S K, Watson R, Nielsen J, Lachapelle G (2005), "Differential Signal Processing Schemes for Enhanced GPS Acquisition", Proc ION-GNSS 2005, Long Beach, 13-16 Sep 2005, pp212-222.



Shanmugam S K, Nielsen J, Lachapelle G, Watson R (2006), “Pre-Correlation Noise and Interference Suppression for Use in Direct-Sequence Spread Spectrum Systems with Periodic PRN Codes”, Proc ION-GNSS 2006, Fort Worth, Sep 2006.