

HEADTAIL Feedback Module: Implementation and Results

J. R. Thompson, *Cornell University, Ithaca, USA*

J. M. Byrd, *LBNL, Berkeley, USA*

W. Höfle, G. Rumolo, *CERN, Geneva, Switzerland*

Abstract

A feedback module has been implemented in the HEADTAIL simulation code in order to investigate the feasibility of a transverse feedback system to damp the electron cloud instability. This instability provokes vertical oscillations within a bunch in the SPS when operated at and above nominal LHC beam current and 25 ns bunch spacing. In the present report the feedback module is described and applied to the case of the SPS at 55 GeV/c, previously identified as worst case energy for the LHC type beams in the SPS in terms of electron cloud instability. The numerical simulations show that a feedback system operating up to 500 MHz with a normalized gain of 0.16 can damp the instability. In contrast, a feedback system only acting on the rigid dipole bunch oscillation cannot cure the high frequency content of the motion.

CERN-AB-2008-070
12 Dec 2008



Geneva, Switzerland

12 December 2008

1 Introduction

In high current proton beams, as accelerated in the SPS, an electron cloud can accumulate in the vacuum chamber. The electron cloud is generated by proton beam induced multipacting initiated by the presence of electrons generated by photoemission or ionization of residual gas. Sufficiently dense electron clouds can lead to beam instabilities in both transverse planes. In dipole magnets these electrons are confined to move in helices in the vertical plane, leading to strong instabilities in this plane. In the SPS, both single and multi-bunch instabilities have been previously observed, particularly in dipole magnets. The results of the simulation code HEADTAIL [1], a program created to study single bunch electron cloud effects, suggest that single bunch electron cloud effects in the SPS are significant only in dipole magnets. This electron cloud related instability can cause significant emittance growth and beam blowup. A summary of observations in the CERN SPS accelerator can be found in [2].

One possible method to control both the single and multi-bunch instability is to implement a feedback system. While the design of a feedback system required to damp transverse multi-bunch instabilities poses certain challenges, similar feedback systems already exist in the SPS and therefore it is believed that implementation of such a system is feasible. More challenging is the implementation of a single bunch feedback system. Such a system may require a very large bandwidth and more than one kick per turn to sufficiently damp the instability, depending on the behavior and growth rate of the single bunch instability. In order to determine the feasibility of such a system we must answer the following questions: What is the worst case instability for typical parameters? Can the instability be damped by kicking just once per turn? What is the minimum bandwidth required to damp the worst case instability? What is the minimum gain required to damp the worst case instability?

In order to answer these questions we utilize an existing tracking code, HEADTAIL. HEADTAIL simulates the interaction between a proton bunch and a uniform electron cloud that has built up inside of the beam pipe [1]. By implementing a module in the existing HEADTAIL code, we were able to simulate the effects of simple feedback on the transverse motion of a single proton bunch. Because we expect single bunch instabilities mainly in the vertical plane, the feedback module only acts on the vertical motion. In Section 3 we discuss the algorithmic structure and simulation results of the implementation of the most basic feedback module: a dipole feedback module. In Section 4, 5 and 6 we discuss the implementation of a more realistic, variable bandwidth feedback module and use it to determine the minimum necessary bandwidth and gain required to damp the instability. Overall, the results of our simple feedback module indicate that damping of the electron cloud induced single bunch instability is possible and realistically achievable for a bandwidth as low as 500 MHz and a normalized gain of $\simeq 0.16$.

2 HEADTAIL Simulation Results

It is important to examine a few typical HEADTAIL simulation results in order to get an idea of the general effects of the electron cloud instability in the SPS. In all simulations we use typical SPS parameter sets for three different momenta: 26 GeV/c, 55 GeV/c and 120 GeV/c. These parameters are summarized in Table 1. Also, we only examine the instability and feedback in the vertical plane, where we expect large electron cloud induced instability. In all three cases the parameters and initial macroparticle distribution were chosen so that the RF voltage is matched. The HEADTAIL algorithm tracks a single bunch by slicing it up into a number of equally spaced slices and tracking the transverse position of each slice at a number of interaction points. As we only expect large electron cloud effects to occur in dipole magnets in the SPS, all ten interaction points are chosen to include only dipole fields. A measure of the transverse oscillation of each slice is given by the “action” weighted by the number of particles in that slice. This quantity is defined as

Momentum	Vertical Beam Size (mm)	rms Bunch Length (m)	Long. Momentum Spread (1σ)
26 GeV/c	2.83	0.206	0.0018
55 GeV/c	1.95	0.217	0.0008
120 GeV/c	1.32	0.184	0.0004

Table 1: Bunch parameters for each energy: For all three energies we used $\beta_y = 72$ m, fractional horizontal and vertical tunes of 0.13 and 0.185, respectively, and a synchrotron tune of 0.003. In all cases the parameters were chosen so that the RF voltage was matched (2 MV). The vertical beam size corresponds to the LHC beam nominal vertical normalized emittance of $3 \mu\text{m}$. The bunch lengths and momentum spread correspond to LHC beam longitudinal emittances of $\simeq 0.35$ eVs, however the bunch length is chosen 25% shorter than the LHC nominal bunch length at injection in order to minimize the quadrupole oscillations and match the injected beam to the 2 MV bucket.

$$Y_{i,j} \equiv N_{i,j} \sqrt{y_{i,j}^2 + \beta_y^2 y'_{i,j}{}^2} \quad (1)$$

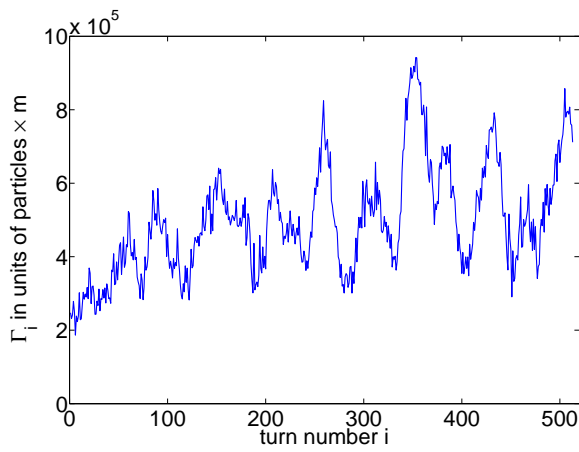
where i is the turn index, j is the slice index, y is the vertical centroid position, y' is the angle of the trajectory of each slice, and $\alpha = 0$ at the interaction points. Examining the growth of the maximum of Y for a bunch over time provides a convenient way to measure the growth of the instability. Therefore, we will typically plot the quantity

$$\Gamma_i \equiv \max [Y_{i,j}] \quad (2)$$

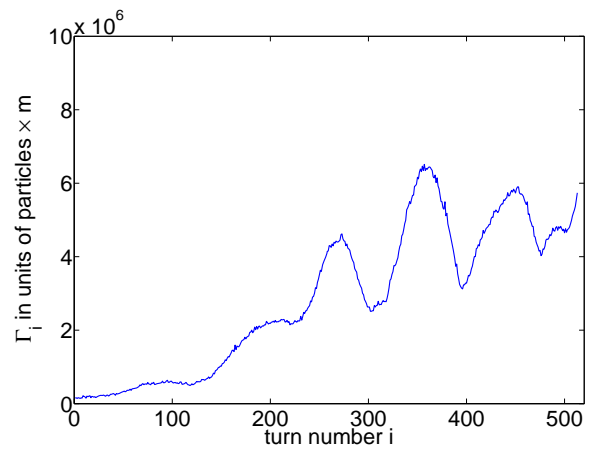
In Fig. 1 we plot Γ_i vs. turn number for all three parameter sets. In the given assumptions (i.e., conservation of emittances and constant rf voltage), the growth rate of the instability is the largest for the 55 GeV/c parameter set. This instability is certainly the most interesting to study in more detail, because it was experimentally observed in the SPS [3]. Besides, 55 GeV/c lies in the range of the candidate values as future injection energy into the SPS with PS2 and the electron cloud instability is the only present SPS limitation that would not benefit from the increase of the injection energy [4]. In this study, we will focus only on determining the minimum upper limits on the gain and bandwidth required to damp the 55 GeV/c case.

In Fig. 2 a) we plot the quantity $Y_{i,j}$ for turns 50 to 150. Before turn 50 one mostly observes noise. That is, the instability really starts to grow only after about turn 50. In this plot a negative time refers to the head of the bunch while a positive time refers to the tail of the bunch. In Fig. 1 a) we observe a very asymmetric behavior; the tail of the bunch has a large amplitude while the head of the bunch does not oscillate. Fig. 1 b) shows Y of each slice over a single bunch and turn. Again we observe asymmetry between the head and tail of the bunch. Both plots suggest that when the instability first emerges, the electron cloud only strongly effects the tail end of the bunch.

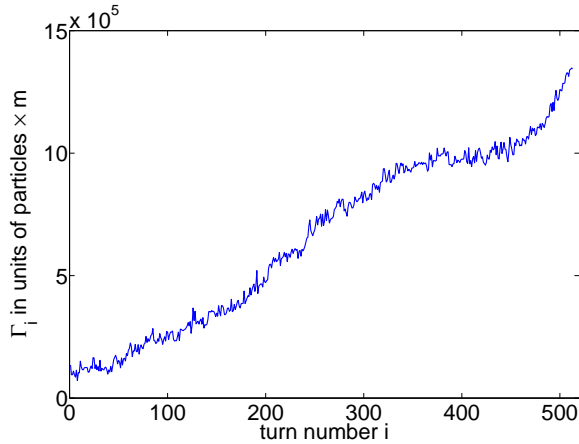
In order to cure this instability a feedback system must have a large enough bandwidth to resolve and damp the asymmetric oscillation. If one decomposes the difference signal into a set of sinusoidal basis functions, the two most dominant modes would be a cosine with period equal to twice the bunch length and a sine with period equal to the bunch length. That is, by superposing those two basis functions one could generate a kick signal that would approximate the asymmetric difference signal. These basis functions would have frequencies of 173 MHz and 345 MHz, respectively. Therefore, an initial hypothesis is that feeding back on these first two ‘‘modes’’ is sufficient and therefore the bandwidth needed to cure the instability is around 345 MHz. But since lower modes can be coupled to higher ones through various mechanisms one may also postulate that feeding back on the average difference signal, or dipole mode, of a bunch may be sufficient to cure the instability. In the following section we will explore this possibility.



(a)

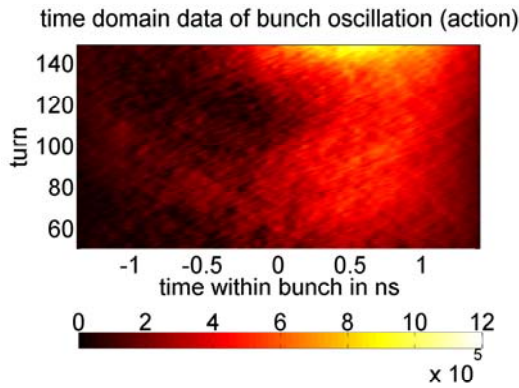


(b)

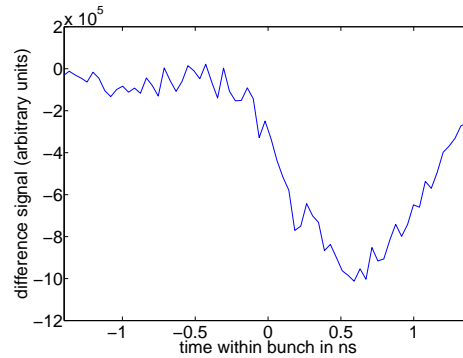


(c)

Figure 1: Γ vs. turn number with no feedback for three energies. a) 26 GeV/c b) 55 GeV/c c) 120 GeV/c. We observe very little growth for 26 GeV/c and the largest growth for 55 GeV/c. Notice the different vertical scales.



(a)



(b)

Figure 2: a) Here Y is plotted vs. turn number (y-axis) and position within bunch (x-axis) with no feedback. Notice the growing asymmetric oscillation within the bunch (i.e. the back of the bunch exhibits oscillation while the front of the bunch is stationary). b) A typical difference signal vs. bunch length plot for 55 GeV/c, in this case, at turn 150. Here we take the difference signal to be proportional to the product of the number of particles in a slice and the centroid position of that slice. Again notice that the back of the bunch has a large transverse offset and the front of the bunch exhibits very little offset.

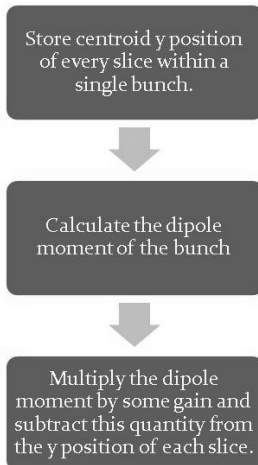


Figure 3: Flow chart for the simple dipole feedback module. This process is repeated once at each of the ten interaction points per turn.

3 Dipole Feedback Module

In order to test the hypothesis that feeding back on the dipole motion of the bunch would be sufficient to cure the instability we have developed a simple dipole feedback module. The HEADTAIL algorithm tracks the transverse position of each slice at ten “interaction points” along the ring. Each of these interaction points is assumed to have identical twiss parameters and $\alpha = 0$. At each interaction point, the dipole feedback algorithm calculates the vertical average offset of the bunch using the vertical centroid slice positions according to

$$y_{\text{dipole}} = \frac{\sum_j N_j y_j}{\sum_j N_j} \quad (3)$$

where y_j is the position of the j_{th} slice and N_j is the number of particles in the j_{th} slice. The quantity $g \times y_{\text{dipole}}$ is then subtracted from the current vertical position of each macroparticle in every slice, where $g \leq 1$ is the normalized gain. The module’s algorithmic structure is summarized in Fig. 3. It is important to note that the dipole feedback module gives an instantaneous position correction. That is, a dipole correction is subtracted from the current position of each particle immediately after each interaction point. While such a feedback is unphysical (i.e. one cannot correct a position instantaneously) it represents a best case scenario; if feeding back on dipole motion will not work using this type of simple feedback, it will not work for any more complicated method.

In Fig. 4 we show the results of dipole feedback with different gains. In a), c) and e) we plot Γ vs. turn number and in b), d) and f) we plot the Fast Fourier Transform of turn 150 (a typical case) for each gain. Notice that plots a) and c) both show growth. While the growth rate of the instability is somewhat smaller than the growth rate without feedback, the instability has by no means been damped. In b) and d) we see that as the gain is increased, the zero frequency (dipole) component decreases. In d) the dipole component is essentially zero. All of this indicates that while feeding back on the dipole removes the dipole component of the bunch difference signal, it is not sufficient to damp higher modes and therefore does not cure the instability. In particular, this type of feedback will not damp modes that have odd symmetry around the bunch center because they have zero dipole component. Earlier we stated that, if we decompose the difference signal into sinusoidal functions, we expect the two most dominant modes to be a cosine function with a period equal to twice the bunch length and a sine function with period equal to the bunch length. Because the cosine function is even we expect it to be damped by simple dipole feedback. But the feedback can do nothing about the sine function. Therefore, in order to have

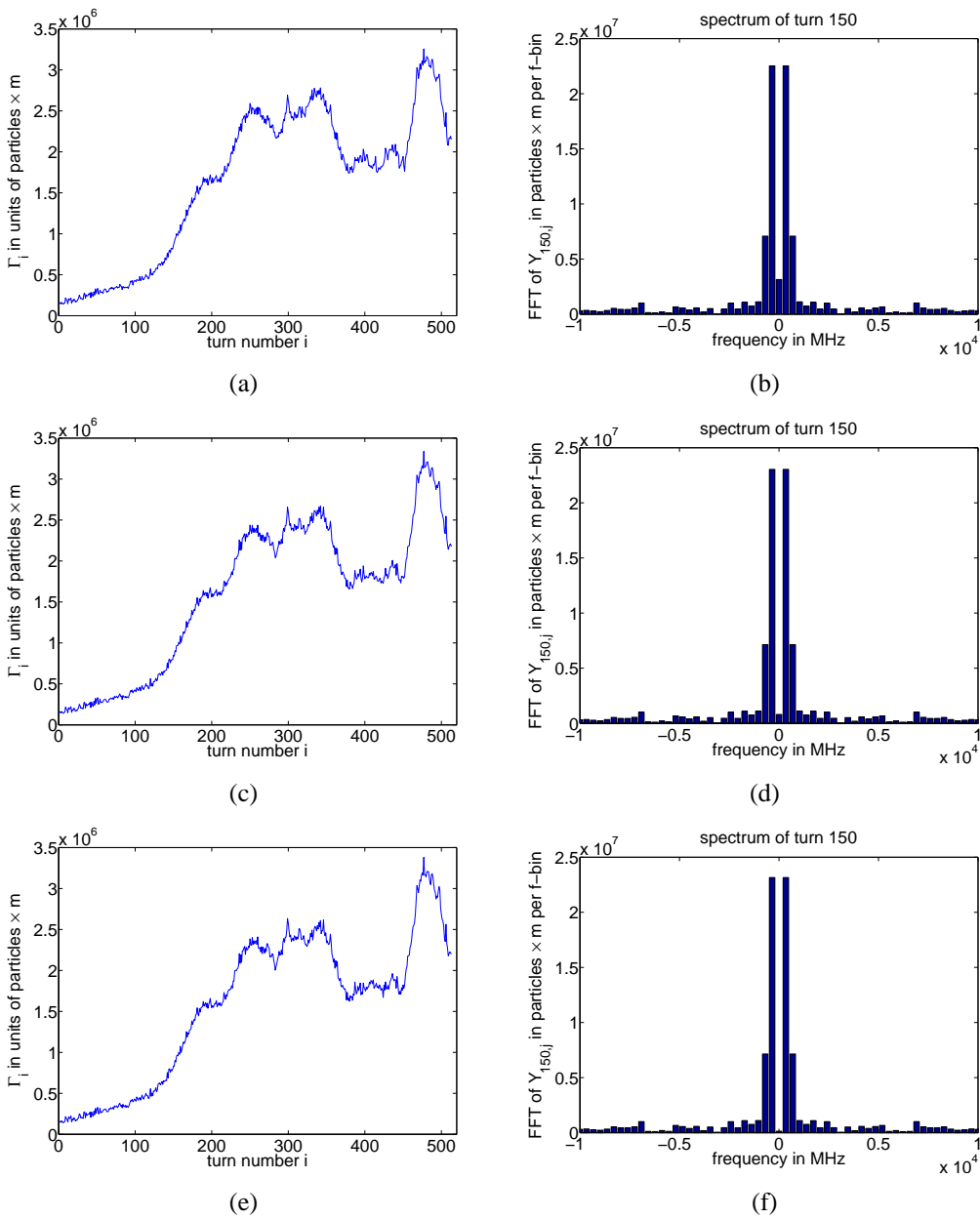


Figure 4: Γ vs. turn number with simple dipole feedback and FFT of turn 150 for three different gain factors. a,b) $1/20$ c,d) $1/10$ e,f) $1/5$. Although the high gain dipole feedback sufficiently removes the dipole component (center frequency of the FFT), it leaves the growth in Γ unchanged. This indicates that curing the instability requires a higher bandwidth than that of the dipole feedback.

any chance of damping the instability we must implement feedback with a wide enough bandwidth to damp both even and odd modes.

4 Variable Bandwidth Feedback Module

In order to have any chance of curing the instability we must implement a feedback that is capable of damping both even and odd modes of different frequencies. Of course, if we decompose the difference signal into sinusoidal modes we expect there to be an infinite number of modes which contribute to a given signal. As developing a feedback system with an infinite bandwidth is impossible we would like to determine the minimum bandwidth of the feedback required to cure the instability. One way to do this is to create a feedback module that allows one to limit the bandwidth of the kick signal delivered to

the bunch. Such a feedback module is presented here.

In this feedback module we not only include a method to limit the bandwidth, but we also use a more physical method; namely we give each macro particle in the bunch a kick once per turn rather than changing its position once per interaction point. In this module both the “pickup” and “kicker” are located at the same point in the ring, at one of the interaction points of the simulation with $\alpha = 0$. The required kick signal can be calculated using the position of each slice at the current and previous turns according to the following formula [5]

$$\Delta y'_{i,j} = g \left[\frac{y_{i,j}}{\beta_y \tan(2\pi q_y)} - \frac{y_{i-1,j}}{\beta_y \sin(2\pi q_y)} \right] \quad (4)$$

where $y_{i,j}$ is the position of the j th slice after the i th turn, q_y is the fractional vertical machine tune, β_y is the vertical beta function at the position of the feedback system and g is a gain factor. In general it is more realistic to feedback on the difference signal rather than the absolute position of each slice. Therefore, we can replace y in the above equation with Ny to get

$$\Delta y'_{i,j} = \frac{g}{\sum_j N_j} \left[\frac{N_j y_{i,j}}{\beta_y \tan(2\pi q_y)} - \frac{N_j y_{i-1,j}}{\beta_y \sin(2\pi q_y)} \right] \quad (5)$$

In order to limit the bandwidth of the kick signal, $\Delta y'_{i,j}$, we filter the signal before “kicking” the bunch. A way of doing this is to implement a moving average filter. The effect of a moving average filter can be described by

$$y_j = \sum_{i=j-(M-1)/2}^{i=j+(M-1)/2} w(i)x_i \quad (6)$$

where y is the filtered data set, x is the unfiltered data set, M is assumed to be an odd integer and known as the window size of the filter and w is the weighting function. There are two important things to notice about this algorithm. First, in order to deal with the “ends” of the data set one must pad the original data with zeros before applying the algorithm. This leads to a filtered data set which is $M-1$ points longer than the original data set. Second, this process is identical to a discrete convolution; the frequency domain function of our filtered data is simply the product of the frequency domain function of the unfiltered data with the Fourier transform of the weighting function. Therefore, we can think of the weighting function as a filtering function and its Fourier transform as the frequency response of the filter. Also, in order to assign a meaningful bandwidth to a given filtering function we must make the assumption that the sampling rate is constant. This translates to having constant slice length over the entire simulation. In general, this is not true in HEADTAIL. HEADTAIL divides the bunch up into a specified number of equal length slices at each interaction point. Therefore, if the bunch length changes during the simulation, the size of each slice will be different at different interaction points. Typically, the bunch length does oscillate due to quadrupole oscillations. To minimize this oscillation we matched the voltage for each parameter set. So, in all cases we examine, the bunch length and slice size are approximately constant. The feedback module has been coded so that it is relatively easy to modify the weighting function and therefore easy to vary the bandwidth of the feedback system. A flow chart of the algorithmic structure of the current feedback system is shown in Fig. 5.

5 Results of Limiting the Feedback Bandwidth

Before limiting the bandwidth of the feedback module, we first must demonstrate that it is possible to cure the instability using a wide bandwidth feedback system. In Fig. 6 we plot Γ vs. turn number for a

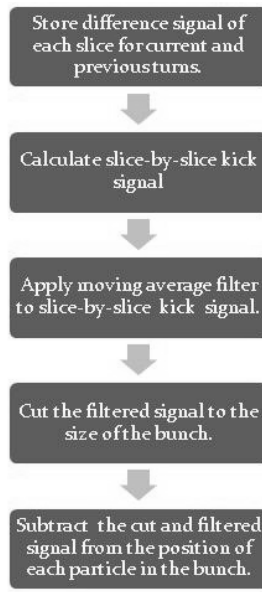


Figure 5: Flow chart for the variable bandwidth feedback module. This process is repeated once per turn.

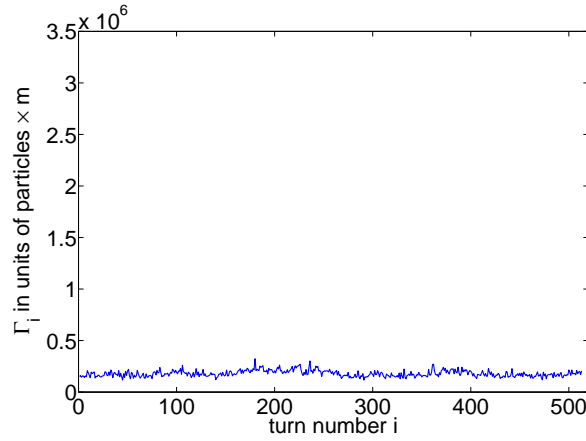


Figure 6: Γ vs. turn number for feedback with the bandwidth filtering turned off. Therefore, the bandwidth of the filtering system is determined by the time per slice. In our case this translates to a bandwidth of 12.6 GHz. Notice that there is no growth, indicating that the instability has been completely cured. The gain factor used is $g=6$.

simulation run with the filtering mechanism turned off. The bandwidth of this feedback system is given by

$$f_{BW} = \frac{1}{2} \frac{N}{\Delta t} \quad (7)$$

where N is the number of slices in the bunch and Δt is the extension of the bunch in time. In this case the bandwidth is 12.6 GHz. Fig. 6 shows Γ vs. turn number for such a high bandwidth feedback system. Notice that we only observe a small noise (no growth). Consequently, we can conclude that the instability can be damped by such a high bandwidth feedback system.

From the discussion in section 3, we expect the minimum bandwidth required to cure the instability to be around 350 MHz. Therefore, we present the results of a bandwidth limited feedback limited to 400 MHz, 300 MHz, and 200 MHz. In all of these simulations, the kick signal was filtered using a windowed sinc filtering function. This filtering function is given by

$$w(i) = K \frac{\sin(2\pi f_c(i - M/2))}{i - M/2} \left\{ 0.42 - 0.5 \cos\left(\frac{2\pi i}{M}\right) + 0.08 \cos\left(\frac{4\pi i}{M}\right) \right\} \quad (8)$$

where f_c is the cutoff frequency, M is the window size, and K is a constant chosen such that $\sum_i w(i) = 1$. Fig. 7 shows Γ vs. turn number plots and the frequency response of the weighting function for each bandwidth as well as the gain factor used. Each simulation was performed multiple times with different gain factors in order to determine the minimum gain factor required to cure the instability for each bandwidth. The plots shown in Fig. 7 are for simulation runs with gain factors near this minimum gain factor. The results show that the minimum bandwidth needed to damp the instability is around 300 MHz.

6 Normalized Gain

All of the gain factors quoted so far have been un-normalized. Typically a normalized gain is defined such that a single kick with a gain of one will fully correct the current y' assuming linear betatron motion. Hence,

$$\Delta y'_{i,j} = g_{\text{norm}} y'_{i,j} \quad (9)$$

In our case the kick can be written as

$$\Delta y'_{i,j} = \frac{g N_{i,j}}{\sum_j N_{i,j}} \left[\frac{y_{i,j}}{\beta_y \tan(2\pi q_y)} - \frac{y_{i-1,j}}{\beta_y \sin(2\pi q_y)} \right] \quad (10)$$

Assuming $\alpha = 0$ at the pickup and purely linear betatron motion one can show that

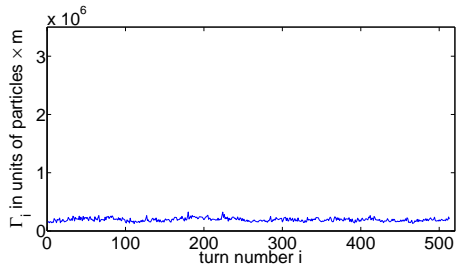
$$y'_{i,j} = \frac{y_{i,j}}{\beta_y \tan(2\pi q_y)} - \frac{y_{i-1,j}}{\beta_y \sin(2\pi q_y)} \quad (11)$$

Therefore, we find

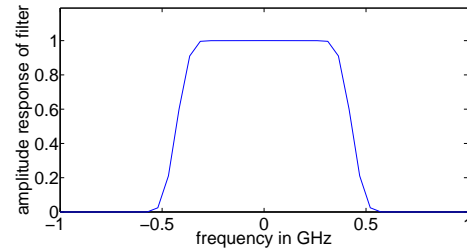
$$g_{\text{norm}} \equiv \frac{g N_{i,j}}{\sum_j N_{i,j}} \quad (12)$$

Hence, the normalized gain effectively changes over the bunch length in proportion to the number of particles in a particular slice. A plot of the quantity $N_{i,j} / \sum_j N_{i,j}$ over the bunch length is shown in Fig. 8.

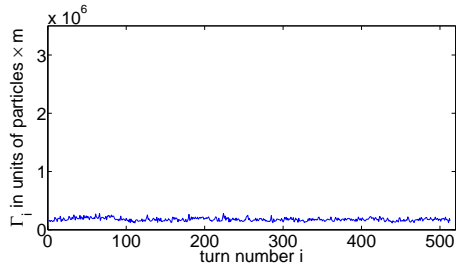
In a real feedback system it is possible to vary the gain over the bunch length. Therefore, a useful quantity to quote is the maximum normalized gain for each turn. But in doing this we do not take into account the asymmetric shape of the difference signal that we are trying to damp. A more appropriate quantity to quote is the normalized gain for the slice that has the maximum difference signal. This represents the actual minimum gain required to damp the instability. Looking at Fig. 2 we see that the maximum difference signal typically occurs for a time of 0.6 ns, which corresponds to a typical multiplicative factor of ≈ 0.016 . In Table 2 we show the normalized gain for the different bandwidth limiting cases that we have looked at in this paper.



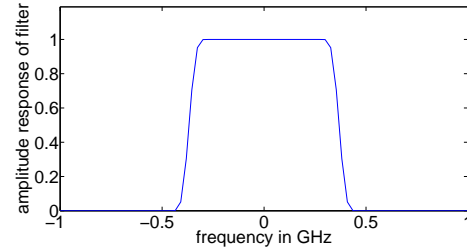
(a)



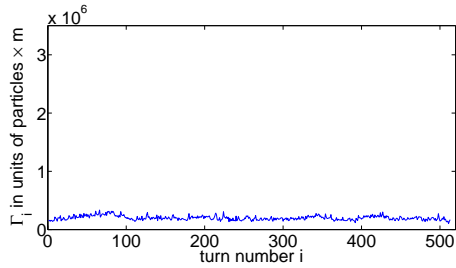
(b)



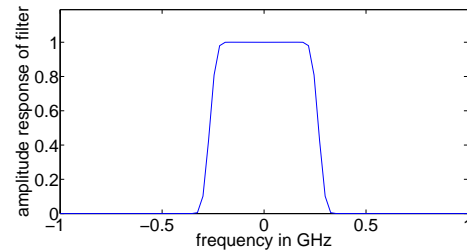
(c)



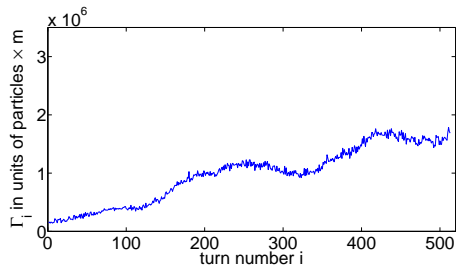
(d)



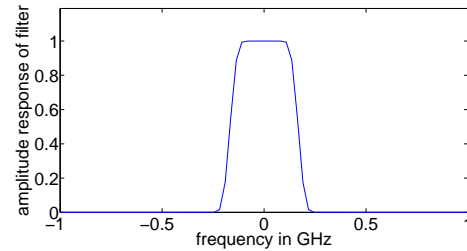
(e)



(f)



(g)



(h)

Figure 7: Here we show the results of limiting the bandwidth of the feedback system. All filtering functions are windowed sinc functions in the time domain: a) Γ vs. turn number with 500 MHz bandwidth limit, $g = 10$ b) frequency response of 500 MHz bandwidth filter c) Γ vs. turn number with 400 MHz bandwidth limit, $g = 20$ d) frequency response of 400 MHz bandwidth filter e) Γ vs. turn number with 300 MHz bandwidth limit, $g = 40$ f) frequency response of 300 MHz bandwidth filter g) Γ vs. turn number with 200 MHz bandwidth limit, $g = 40$ h) frequency response of 200 MHz bandwidth filter

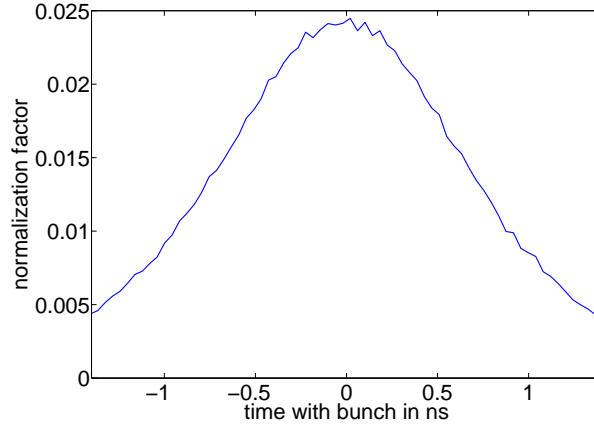


Figure 8: Plot of $N_{i,j} / \sum_j N_{i,j}$ vs. slice position within the bunch. This is the multiplicative factor required to normalize the gain. As a typical asymmetric bunch oscillation has a maximum “difference signal” at 0.6 ns, we conclude that the minimum gain required to damp oscillations is given by $0.016 \times g$.

Minimum Gains for Various Bandwidths		
Bandwidth	Gain Factor	Normalized Gain
12.6 GHz	6	0.096
500 MHz	10	0.16
400 MHz	20	0.32
300 MHz	40	0.64

Table 2: List of the minimum gain factor and normalized gain required to cure the instability using feedback with different bandwidths.

7 Conclusion

Our studies with the HEADTAIL feedback module have focused on examining the behavior of the worst case instability (the 55 GeV/c case) in the SPS. We found that in order to cure this instability one must have a large enough bandwidth feedback system to handle the asymmetric oscillation in the bunch difference signal. From a simple analysis we expected that the minimum bandwidth required to damp this oscillation is around 350 MHz. By implementing a bandwidth limiting feedback module we were able to determine that the minimum bandwidth required to cure the instability is actually around 300 MHz. But the normalized gain of such a relatively low bandwidth feedback system is relatively high. We also established a lower limit on the normalized gain required by determining the minimum gain required to cure the instability for a large bandwidth feedback system. This was $\approx 1/10$. A feasible gain is typically less than about $1/5$. According to our studies, the lowest bandwidth for which this gain limit can be achieved is about 500 MHz. Therefore, we expect that a feedback system with 500 MHz bandwidth and gain of about 0.16 would be the most realistically realizable system. The feedback module can now be used to study a range of feedback and beam parameters for the SPS as well as other accelerators as the PS2 with injection at 50 GeV/c. It is also planned to extend the feedback module to cover an accelerating bucket and optimize the signal processing (delay errors and adaptation) to realistically cover the accelerating ramp.

Acknowledgments

The authors would like to J. Fox for very good discussions on feedback systems and G. Arduini for his initiative to start the project and helpful comments.

References

- [1] G. Rumolo, F. Zimmermann, "Electron Cloud Simulations: beam instabilities and wakefields", *Phys. Rev. Spec. Top. Accel. Beams* **5**, 121002 (2002)
- [2] G. Arduini, T. Bohl, K. Cornelis, W. Höfle, E. Métral, F. Zimmermann, "Beam Observations with Electron Cloud in the CERN PS & SPS Complex", 31st Advanced ICFA Beam Dynamics Workshop on Electron Cloud, Napa, CA USA, 19-23 Apr 2004, CERN-2005-001, 31-47, CERN, Geneva (2005)
- [3] G. Rumolo, G. Arduini, E. Métral, E. Shaposhnikova, E. Benedetto, R. Calaga, G. Papotti, B. Salvant, "Dependence of the Electron-Cloud Instability on the Beam Energy", *Phys. Rev. Lett.* **100**, 144801 (2008)
- [4] G. Rumolo, E. Métral, E. Shaposhnikova, "SPS performance with PS2", in *Proc. of LHC LUMI 2006 CARE-HHH-APD Workshop*, Valencia, Spain, 16-20 Oct 2006, CERN-2007-002, 129-134, CERN, Geneva (2007)
- [5] J. M. Byrd, "Simulations of the PEP II Transverse Coupled-bunch Feedback System", *Proceedings of the 16th IEEE Particle Accelerator Conference (PAC 95)*, Dallas, TX (USA), 1-5 May 1995 vol.4, 2684-2686 (1995)