Dynamic Electron Cloud Modeling

Sumner Hearth

May 9, 2016

1 Goal

The goal of this project is to produce a bmad plugin that correctly emulates the bunch growth and tune shifts seen in experiments using a dynamic electron cloud. The electron cloud is modeled as a gaussian charge distribution with dynamic position $\overrightarrow{\mu} = (\mu_x, \mu_y)$ and total charge q.

For now all the computation described below is only applied to lattice elements representing dipole bends.

2 Modeling

2.1 Static Cloud

Consider first a static cloud with no bunch growth, translation, or pinching. The cloud in this scenario is modeled as a gaussian distribution with parameters q, σ_x, σ_y, referring to *total charge*, and x and y variance respectively. The program is written into a Bmad `track1_postprocess` module which means it is applied to each particle at the end of each lattice element. As a particle enters we determine whether the particle belongs to a new bunch, or is another particle in the current bunch. This is determined based on the current time of the particle versus the time of the last seen particle. If that time exceeds half the expected inter-bunch time ($\frac{14ns}{2}$) then it is considered to be a new bunch. The kick applied to the particle is then calculated using bmad's `bbi_kick` procedure. We supply to it the relative distance of the particle (from `end_orb`) and the $\frac{\sigma_x}{\sigma_y}$ ratio. This gives us a relative kick value in either dimension which we must scale by a factor `bbi_const`. This constant is determined by $\frac{q}{(2\pi\gamma(\sigma_x+\sigma_y))}$. This is not directly as stated in the documentation, which states

bbi_const = N_particles_bunch * e_charge / (2 * pi * gamma * (sig_x + sig_y))

However so long as it is maintained that the charge `q = N_particles_bunch * e_charge` then this should be fine. This constant times the relative kick values gives the change in momentum in radians, which is added to the particle passing through the lattice element.

For a particle with given $(\overrightarrow{x}, \overrightarrow{p})$ and cloud $(\overrightarrow{\mu}, \overrightarrow{\sigma})$ in a lattice element with length (s) we update the particle characteristics as follows:

$$\overrightarrow{k} = \text{bbi\_kick}\left(\frac{x-\mu_x}{\sigma_x}, \frac{y-\mu_y}{\sigma_y}, \frac{\sigma_x}{\sigma_y}\right)$$

$$\overrightarrow{p}_{i+1} := \overrightarrow{p}_i - \overrightarrow{k}\left(\frac{q}{2\pi\gamma(\sigma_x+\sigma_y)}\right) s$$

2.2 Moving Cloud

In order to calculate the expected position change of the cloud, the whole gaussian carries with it an accumulated momentum $\overrightarrow{p} = (p_x, p_y)$. If the particle is determined to be part of a new bunch, the momentum and position of the electron cloud are reset to 0, then calculations proceed as normal. If, however, the particle is a continuation of the current bunch, then we take the time difference between the previous bunch and the current, multiply that by $\overrightarrow{p}$, then divide by the total mass of the cloud (given by $m_c = \frac{qm_e}{e^-}$). This gives the first order approximation for the change in position for the cloud[1]. After the particle kick has been calculated, the same kick times the reference energy of the particle (in order to get absolute momentum change instead of angular) is subtracted from the total momentum of the cloud, though it is also multiplied by the relative charge of the macro-particle versus the charge of an ordinary positron to account for the assumptions made in `bbi_kick`. This new $\overrightarrow{p}$ is saved for the next particle interaction.

For a particle with given (x, y, p_0) in a lattice element with length (s) we update the cloud characteristics $(\overrightarrow{\mu}_i, \overrightarrow{p}_i, \overrightarrow{\sigma})$ as follows:

$$\overrightarrow{\mu}_{i+1} := \overrightarrow{\mu}_i + \overrightarrow{p}_i \left(\frac{e^-}{qm_e}\right) c\Delta t$$

$$\overrightarrow{k} = \text{bbi\_kick}\left(\frac{x - [\overrightarrow{\mu}_i]_x}{\sigma_x}, \frac{y - [\overrightarrow{\mu}_i]_y}{\sigma_y}, \frac{\sigma_x}{\sigma_y}\right)$$

$$\Delta\overrightarrow{p} = \overrightarrow{k}\left(\frac{q}{2\pi\gamma(\sigma_x + \sigma_y)}\right) s$$

$$\overrightarrow{p}_{i+1} := \overrightarrow{p}_i + \Delta\overrightarrow{p}\left(\frac{q}{e^-}\right) p_0$$

2.3 Cloud Pinching

Cloud pinching changes the effective charge of the cloud during the passage of a bunch. It uses the z position of the particle (which is relative to the center of the particle bunch) to scale the charge q' of the cloud between q and $(\alpha + 1)q$ for some parameter α. It approximates this growth using a gaussian centered about the end of the bunch ($z = \sigma_z$):

$$q' := q\left(1 + \alpha e^{-\left(\frac{z}{\sigma_z}+1\right)^2}\right)$$

This new q' is then substituted into any of the previous equation where q would be used.

This increase in the effective density is to match the increase seen in ecloud simulations[2]

2.4 Feedback

The executable employed (`multibunch`) will reset the x and y position and momentum at the end of each turn. The center of mass on each dimension may be calculated (assuming all particles are weighted proportionally to their charge), and then subtracted from each respective dimension. The amount that is subtracted is scaled by the strength of the feedback (0 is no feedback, 1 is full reset). This setting is contained in the `beam_def.in` file

3 Results

3.1 Cloud Effects

The bunch passing through the cloud will generate emittance growth dependent on the density of the cloud. The densities are based on the slides from Prof. Crittenden indicating the density in the chamber near the cloud is on the order of10^{12} $e^-.m^{-3}$.

Plotted in Figure 2 is the bunch growth on the vertical axis of the beam as it passes through a moving cloud with no pinch, and with feedback fully enabled. The the subsequent plots Figure 3 and Figure 4 we

[1]Could instead implement a runge-kutta style approximation, but difficult due to execution order of particles.
[2]`https://wiki.classe.cornell.edu/pub/ILC/Private/CesrTA/CollabMeetings/EC_WB_STP_20151201.pdf`

see the same strength cloud but with pinching enabled at a values seen in simulations. There is massive bunch growth for the $10^{13} e^{-} m^{-3}$ cloud. But smaller than without pinching, which had caused the bunch to disappear.

3.2 Varying Initial Offset

By varying the initial offset of the bunch we can cause the vertical size to increase. Increases in the cloud density slightly mitigate this effect. See Figure 5

3.3 Varying Feedback Strength

Varying feedback strength is almost identical to turning feedback completely on or completely off. Feedback strength above 5% is indistinguishable from 100%. See Figure 6 and Figure 7 for simulations with dynamic clouds.

3.4 Varying Pinching

By varying the strength of pinching I found bunch size growth, though with this model the growth only appears at large pinch strengths ($\alpha = 4 - 5$). See Figure 8 through Figure 10

4 Simulation conditions

This simulations have all been performed using the `cta_2085mev_xr40m_20101215_bs.lat` lattice. Radiation damping is enabled.

5 Known Issues

The main issue I've run into with the physicality of the system is that the electron cloud may be accelerated past the speed of light. Over the course of the passage of the bunch of positrons the cloud can move further than the bunch, indicating a problem. This is likely a result of the cloud position not getting reset under certain conditions (since resets are based on how long it has been since a particle was seen and not the actual turn number).

6 Plots

Appendix section for plots.

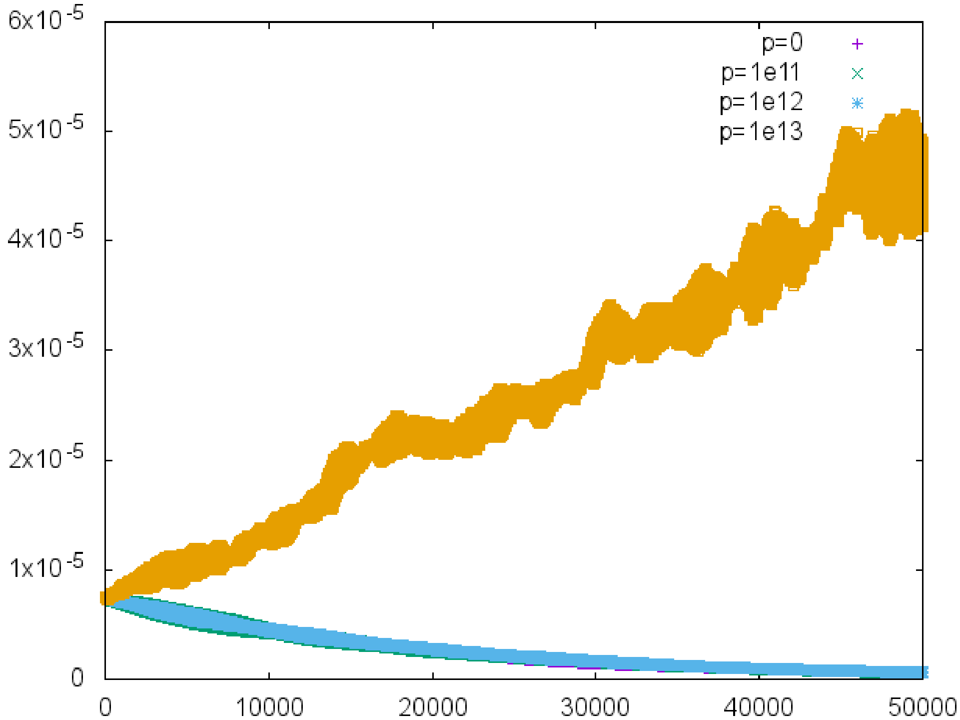

Figure 1: Vertical size with varying density, no feedback

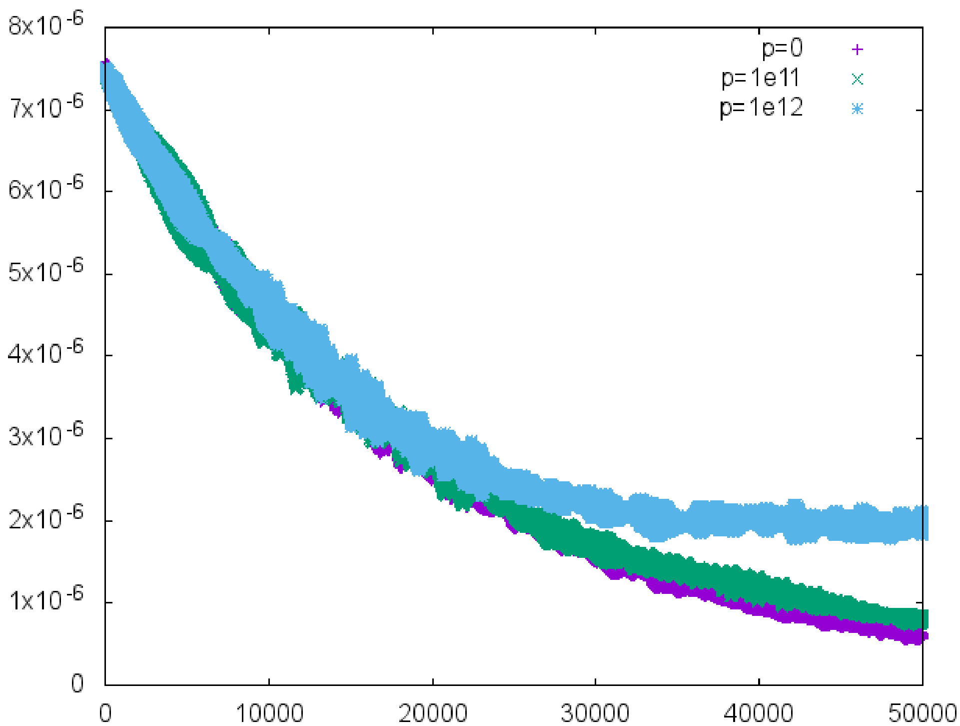

Figure 2: Vertical size with varying density, feedback (10^{13} dead)

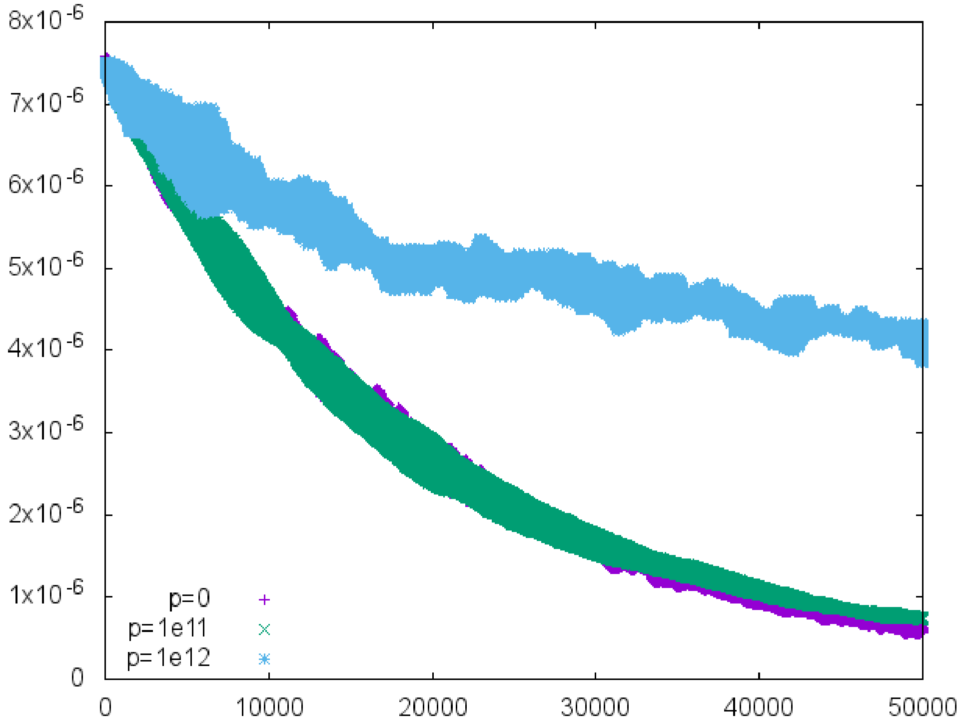

Figure 3: Vertical size with varying density, small pinching ($\alpha = 1$)

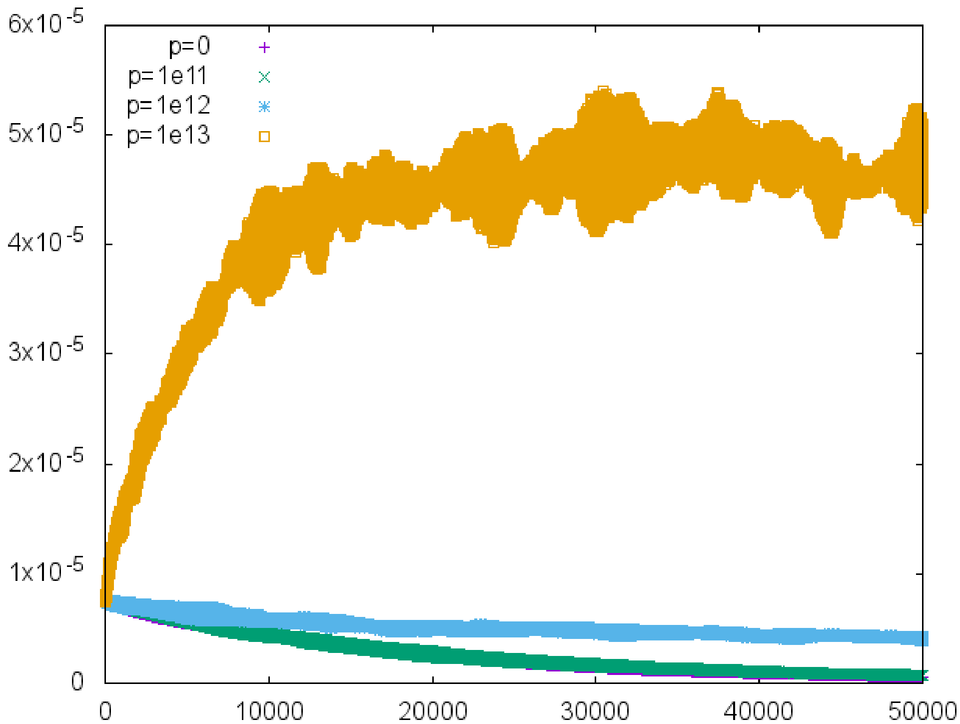

Figure 4: Varying density, small pinching ($\alpha = 1$), $\rho = 10^{13}$ included

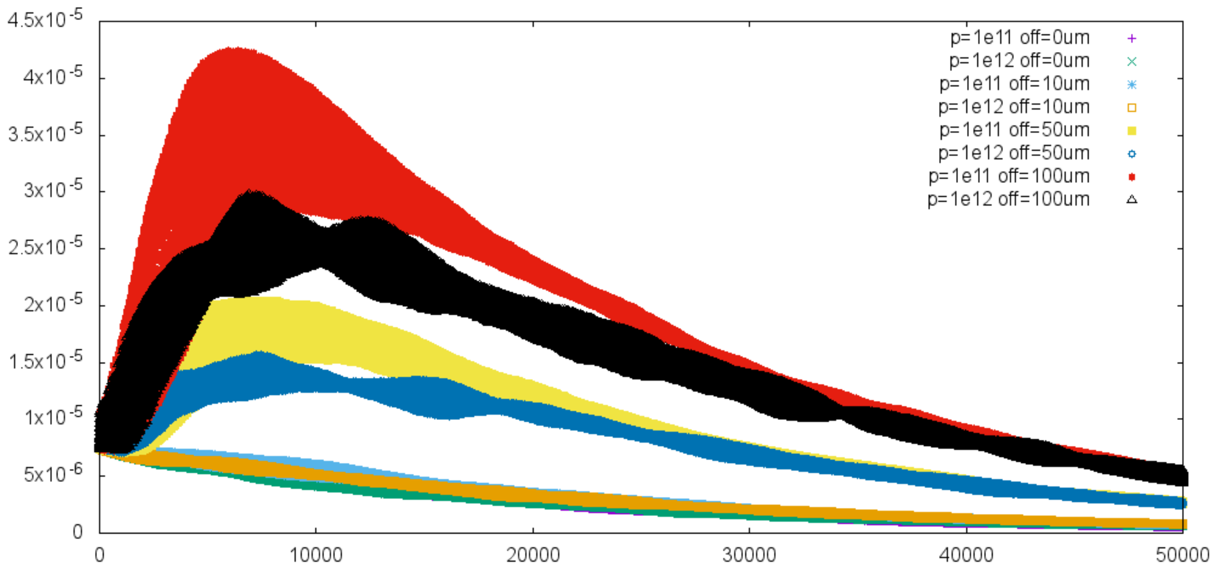

Figure 5: Vertical size for varying initial offsets

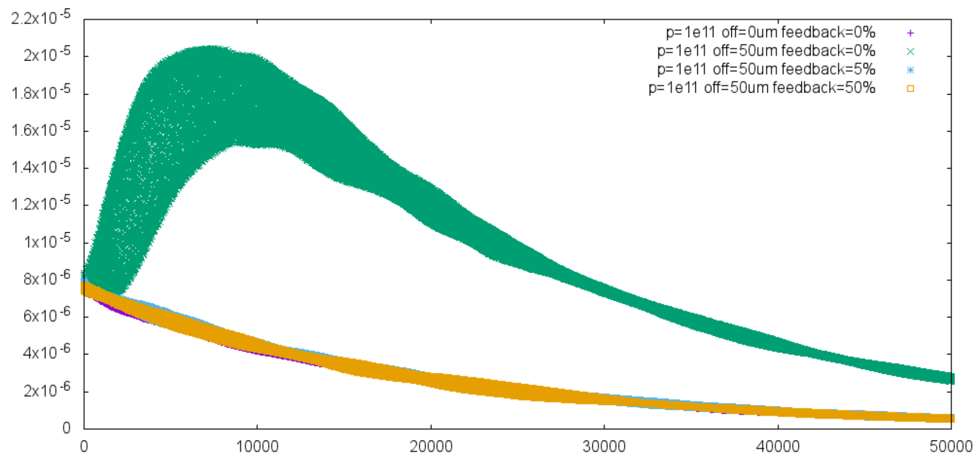

Figure 6: Vertical size with varying feedback for $\rho = 10^{11}$, small pinching ($\alpha = 1$)

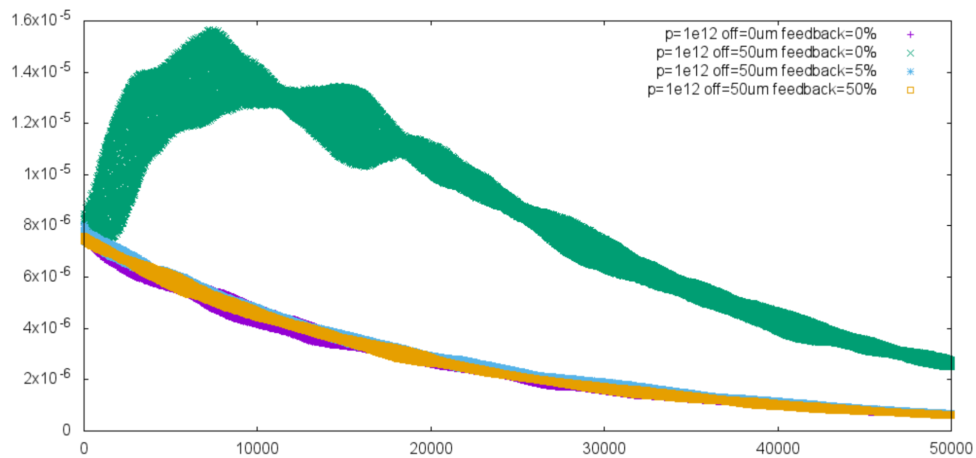

Figure 7: Vertical size with varying feedback for $\rho = 10^{12}$, small pinching ($\alpha = 1$)

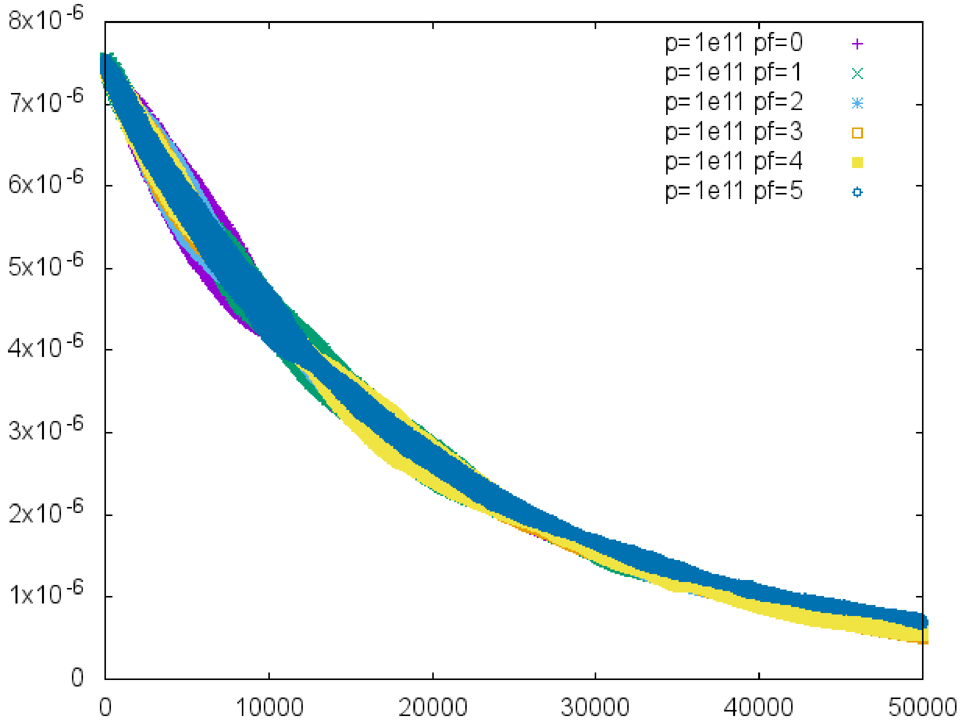

Figure 8: Vertical size with varying pinching, no feedback

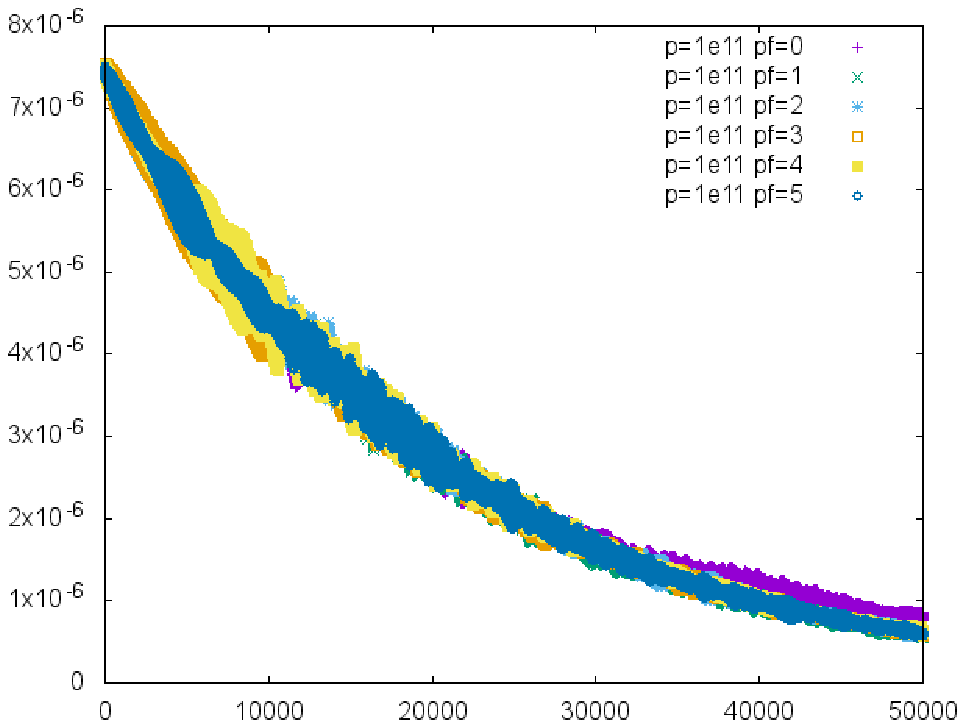

Figure 9: Vertical size with varying pinching, feedback

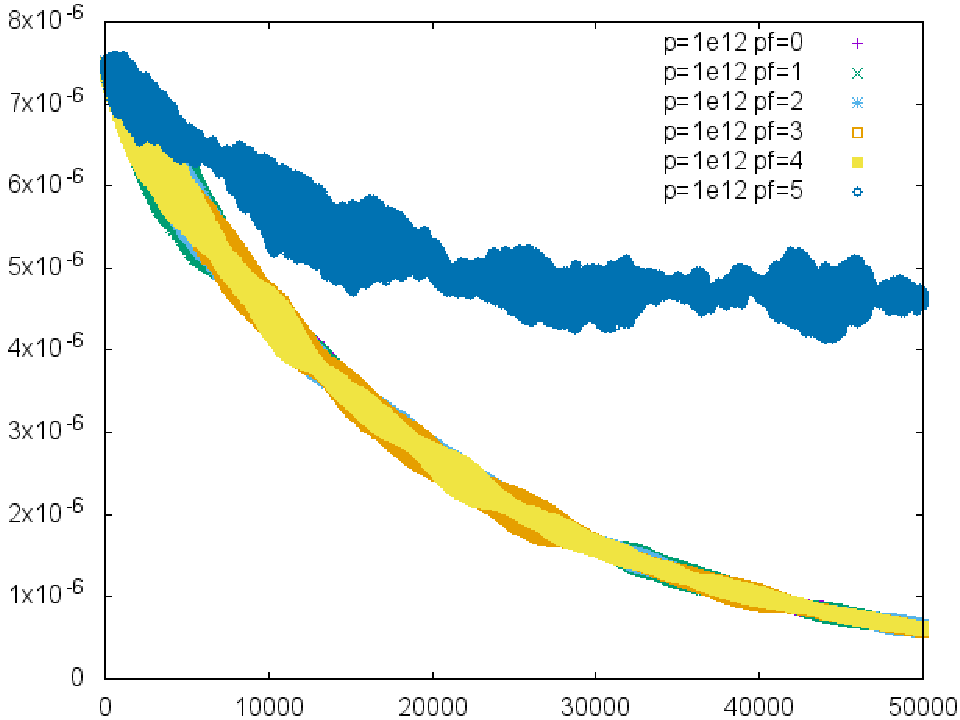

Figure 10: Vertical size with varying pinching, no feedback

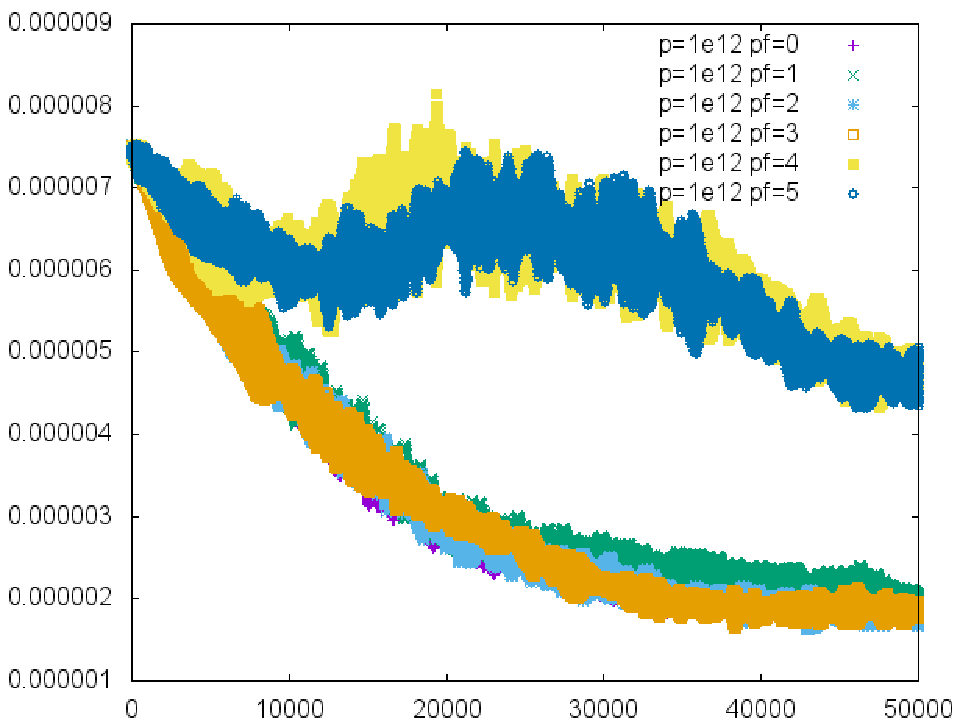

Figure 11: Vertical size with varying pinching, feedback