BNL-105169-2014-TECH

Booster Technical Note No. 125;BNL-105169-2014-IR

RANDOM SEXTUPOLE CORRECTION

S. Tepikian

August 1988

Collider Accelerator Department

Brookhaven National Laboratory

U.S. Department of Energy

USDOE Office of Science (SC)

RANDOM SEXTUPOLE CORRECTION

AD

Booster Technical Note

No. 125

S. TEPIKIAN

AUGUST 5, 1988

ACCELERATOR DEVELOPMENT DEPARTMENT

Brookhaven National Laboratory

Upton, N.Y. 11973

RANDOM SEXTUPOLE CORRECTION

S. Tepikian

Abstract

The AGS-Booster has a large tune spread and strong random eddy current sextupoles at injection. As a result, particles in the beam may cross four strong imperfection resonances. We correct these four resonances using trim coils in the chromaticity sextupoles. A scheme is presented for determining the strength of these correctors. For random sextupole errors of 10% of the systematic eddy current sextupoles in the dipoles and .1% errors in the chromaticity sextupoles, we find that the maximum pole tip field for these trim coils is about 35 Gauss.

1. Introduction

The AGS-Booster is designed to be a fast cycling machine to increase the space charge limit of the AGS. Hence, at injection, the dipoles contain large eddy current sextupoles due to the fast cycling[1] and the space charge tune spread is very large. The random component of the eddy current sextupoles can excite imperfection resonances that may be crossed by the space charge tune spread.

The imperfection resonances that are important can be seen from the tune diagram in Fig. 1. Only the resonances excited by sextupoles are shown in this figure. The dotted line shows the possible range of tune spread due to the space charge of the beam. The four imperfection resonances that can be crossed are

$$3\ \nu_x = 14$$

$$\nu_x + 2\ \nu_y = 14$$

$$3\ \nu_x = 13$$

$$\nu_x + 2\ \nu_y = 13.$$

During machine operation, it may be found that some of the other nearby resonances are also important (such as $\nu_x - 2\ \nu_y = -4$, etc.). These can be included but with some additional expense: (1) stronger correctors and (2) a larger control program to determine the corrector strengths may be required.

The imperfection resonances listed above can be canceled by including sextupole correctors in the chromaticity sextupoles. These imperfection resonances are assumed to be excited by random sextupoles. There are two sources of random sextupoles considered: (1) from eddy currents in the dipoles at injection and (2) from errors in the chromaticity sextupoles.

We start with the theory given in the next section. Section 3 gives the strategy we used in finding the corrector strengths. Finally, a conclusion is given in section 4.

2. Theory

The motion of a particle in the beam of an accelerator can be described by the following Hamiltonian

$$H = \frac{p_x^{\ 2}}{2} + \frac{p_y^{\ 2}}{2} + \left[K(s) + \frac{1}{\rho^2(s)}\right] \frac{x^2}{2} - K(s)\ \frac{y^2}{2} + \frac{S(s)}{6}\ (x^3 - 3\ x\ y^2)$$

where x and y are the particles position with respect to the equilibrium, p_x and p_y are the conjugate momenta, K(s) is the quadrupole strength, S(s) is the sextupole strength, $\rho(s)$ is the radius of curvature and s is the independent variable (i.e. the time variable). We have assumed the particle is at the equilibrium momentum.

The resonance strengths can be found by applying two canonical transformations to the Hamiltonian H and then expanding the potential term in a fourier series. The first canonical transformation (which transforms the Hamiltonian to action-angle[2] variables) is given by the generating function shown below:

$$F(x,y,\psi_x,\psi_y,s) = -\frac{x^2}{2\beta_x(s)}\left[\tan\psi_x - \frac{\beta'_x(s)}{2}\right] - \frac{y^2}{2\beta_y(s)}\left[\tan\psi_y - \frac{\beta'_y(s)}{2}\right]$$

where ψ_x and ψ_y are the new angle variables, primes denote d/ds and the functions $\beta_x(s)$ and $\beta_y(s)$ are the solutions of the following equations

$$\frac{\beta_x(s)\beta''_x(s)}{2} - \frac{(\beta'_x(s))^2}{4} + \left[K(s) + \frac{1}{\rho^2(s)}\right]\beta_x^2(s) = 1$$

$$\frac{\beta_y(s)\beta''_y(s)}{2} - \frac{(\beta'_y(s))^2}{4} - K(s)\,\beta_y^2(s) = 1.$$

The actions, J_x and J_y, are related to x and y as follows

$$x = \sqrt{2J_x\beta_x(s)}\cos\psi_x$$

$$y = \sqrt{2J_y\beta_y(s)}\cos\psi_y.$$

Note, the emittance of the beam is related to the action by $2J_x = E_x/\pi$ and $2J_y = E_y/\pi$. If there are no sextupoles present (i.e. S(s) = 0) then the emittances (and the actions) are invariants of the motion[3].

Using the generating function, F, the new Hamiltonian becomes

$$H_1 = \frac{J_x}{\beta_x(s)} + \frac{J_y}{\beta_y(s)} + \frac{S(s)}{3}\sqrt{2J_x\beta_x(s)}\,(J_x\beta_x(s)\cos^2\psi_x - 3J_y\beta_y(s)\cos^2\psi_y)\cos\psi_x.$$

The next transformation will eliminate the time dependence in the first two terms in the Hamiltonian H_1. This transformation is given by the following generating function

$$G(I_x, I_y, \psi_x, \psi_y, s) = I_x[\psi_x - \mu_x(s)] + I_y[\psi_y - \mu_y(s)]$$

where I_x and I_y are the new action variables and

$$\mu_x(s) = \int_0^s \frac{dt}{\beta_x(t)} - \frac{2\pi}{C} \nu_x s$$

$$\mu_y(s) = \int_0^s \frac{dt}{\beta_y(t)} - \frac{2\pi}{C} \nu_y s$$

and

$$\nu_x = \frac{1}{2\pi} \int_0^C \frac{dt}{\beta_x(t)} ,$$

$$\nu_y = \frac{1}{2\pi} \int_0^C \frac{dt}{\beta_y(t)}$$

with C being the circumference of the accelerator. Note, $\mu_x(s)$ and $\mu_y(s)$ are periodic functions of s with a period of C/P where P is the periodicity of the lattice.

The new action angle variables are related to the old as follows

$$I_x = J_x$$

$$I_y = J_y$$

$$\xi_x = \psi_x - \mu_x(s)$$

$$\xi_y = \psi_y - \mu_y(s)$$

where ξ_x and ξ_y are the new angle variables.

The transformation of Hamiltonian H_1 with the generating function G leads to the following Hamiltonian

$$H_2 = \frac{2\pi}{C}\left[\nu_x I_x + \nu_y I_y + [2I_x]^{3/2}\sum_{n=-\infty}^{\infty} e^{-i2\pi ns/C}\,(A_n\, e^{i3\xi_x} + \right.$$

$$+ 3B_n e^{i\xi_x}) - [2I_x]^{1/2}[2I_y]\sum_{n=-\infty}^{\infty} e^{-i2\pi ns/C}\,(C_n e^{i(\xi_x+2\xi_y)} +$$

$$\left. + D_n e^{i(\xi_x-2\xi_y)} + 2E_n e^{i\xi_x})\right]$$

where

$$A_n = \frac{i}{48\pi}\int_0^C S(t)\beta_x^{3/2}(t)e^{i[3\mu_x(t)+2\pi nt/C]}\,dt$$

$$B_n = \frac{i}{48\pi}\int_0^C S(t)\beta_x^{3/2}(t)e^{i[\mu_x(t)+2\pi nt/C]}\,dt$$

$$C_n = \frac{i}{16\pi}\int_0^C S(t)\beta_x^{1/2}(t)\beta_y(t)e^{i[\mu_x(t)+2\mu_y(t)+2\pi nt/C]}\,dt$$

$$D_n = \frac{i}{16\pi}\int_0^C S(t)\beta_x^{1/2}(t)\beta_y(t)e^{i[\mu_x(t)-2\mu_y(t)+2\pi nt/C]}\,dt$$

$$E_n = \frac{i}{16\pi}\int_0^C S(t)\beta_x^{1/2}(t)\beta_y(t)e^{i[\mu_x(t)+2\pi nt/C]}\,dt.$$

The terms A_n, B_n, C_n, D_n and E_n give the resonance strengths for the following resonances

$$3\,\nu_x = n$$

$$\nu_x = n$$

$$\nu_x + 2\nu_y = n$$

$$\nu_x - 2\nu_y = n$$

$$\nu_x = n$$

respectively. In the next section we will minimize the resonance strengths for the resonances given in the introduction.

3. Strategy

The AGS-Booster consist of six superperiods and each superperiod contains eight chromaticity correcting sextupoles. Each chromaticity sextupole will have a trim coil which will be used to correct for random sextupole errors. Using these trim coils we want to generate a sextupole resonance that cancels those resonances excited by the random sextupoles. Thus, in an ideal accelerator with no random sextupole errors, we want to find the strengths of the trim coil sextupoles to excite any given imperfection resonance.

We are concerned with the four resonances listed in the introduction. The corresponding strengths are A_{14}, C_{14}, A_{13} and C_{13}. Since these numbers are complex, there are eight conditions to be satisfied. The integrals in the previous section relating the sextupole strengths to the resonance strengths can be converted to sums by using the thin lens approximation as

$$A_n = \frac{i}{48\pi} \sum_{p=1}^{6} \sum_{q=1}^{8} S_{pq} \beta_x^{3/2}(t_{pq}) e^{i[3\mu_x(t_{pq})+2\pi n t_{pq}/C]}$$

$$C_n = \frac{i}{16\pi} \sum_{p=1}^{6} \sum_{q=1}^{8} S_{pq} \beta_x^{1/2}(t_{pq}) \beta_y(t_{pq}) e^{i[\mu_x(t_{pq})+2\mu_y(t_{pq})+2\pi n t_{pq}/C]}$$

where the outer sum is over the superperiods, S_{pq} are the integrated sextupole strength of the correctors and t_{pq} are the positions of the center of the correctors.

Due to the periodicity of the β and μ functions the above sums can be simplified. First, we define

$$t_q = t_{(p=1)q}$$

then

$$t_{pq} = t_q + (p-1)C/6$$

for p = 1, 2, 3, ... 6 and q = 1, 2, 3, ... 8. Defining the coefficients f_p so that

$$S_{pq} = f_p S_q$$

relates the sextupole strength of the correctors from one superperiod to the next. The sums can now be written as

$$A_n = \frac{i}{48\pi}\left[\sum_{p=1}^{6} f_p e^{i\pi n(p-1)/3}\right]\sum_{q=1}^{8} S_q \beta_x^{3/2}(t_q) e^{i[3\mu_x(t_q)+2\pi n t_q/C]}$$

$$C_n = \frac{i}{16\pi}\left[\sum_{p=1}^{6} f_p e^{i\pi n(p-1)/3}\right]\sum_{q=1}^{8} S_q \beta_x^{1/2}(t_q)\beta_y(t_q) e^{i[\mu_x(t_q)+2\mu_y(t_q)+2\pi n t_q/C]}$$

Choosing $f_p = \cos[\pi n(p-1)/3]$ for n = 13 and for n = 14 leads to the greatest contribution to each of the two sets of resonances separately. Furthermore, the two sets of f_p for n = 13 and for n = 14 are orthogonal (i.e. f_p of n = 13 doesn't contribute to A_{14} and C_{14} and vice versa).

Now we can solve for the unknowns S_q for a given choice of A_{13}, C_{13}, A_{14} and C_{14}. Due to the orthogonality between n = 13 and n = 14 we have sixteen unknowns (from two independent sets of S_q) with eight conditions. These additional unknowns are handeled by the following scheme:

$$S_1 = S_5\ ,\quad S_2 = S_6\ ,\quad S_3 = S_7 \quad \text{and} \quad S_4 = S_8$$

A listing of a program that uses the above strategy for solving for the trim coil sextupole strengths is given in the appendix. Using this program we have calculated the sextupole correctors for various different random seeds of sextupole errors. We have assumed a 10% variation on the systematic value of the eddy current sextupole (i.e. B'' = .24 T/m^2) in the dipoles and 0.1% error in the chromaticity sextupoles. In each case we used a uniform distribution of errors. The table below shows the random number seed versus the maximum strength of the corrector

SEED	MAXIMUM CORRECTOR STRENGTH (m^{-2})	POLE TIP FIELD (GAUSS)
78687	.0502	34.5
19163	.0140	9.6
8104411	.0113	7.8
47887	.0499	34.3
838301	.0430	29.6

where we assumed a trim coil 10 cm long and the pole tips 8 cm from the axis.

Figure 2 shows how the pole tip field strength of the trim coils change when the strength of the random sextupole component changes. This was calculated with a random seed of 78687.

4. Conclusion

A particle in the AGS-Booster can cross four strong imperfection resonances due to the large space charge tune spread at injection. We have formulated a strategy for correcting these resonances using a set of trim coils in the chromaticity sextupoles.

In order to design this system we must know the largest possible pole tip fields required by the trim coils. To do this we ran several test cases using different random seeds. The results of these runs are given in the above table. We find the maximum pole tip field to be about 35 Gauss when the random sextupoles are from 10% of the systematic eddy current sextupoles in the dipoles. The effect of the 0.1% errors in the chromaticity sextupoles are negligable. If more resonances than the four given in the introduction are required to be corrected, then the upper limit may need to be increased.

References

1. G. Morgan and S. Kahn, Booster Tech Note #4

2. H. Goldstein, Classical Mechanics, (Addison Wesley, 1965)

3. E. D. Courant and H. S. Snyder, Annals of Physics 3, 1-48 (1958)

4. W. H. Press, et al, Numerical Recipes (Cambridge University Press, 1986)

Appendix

Below is a listing of the program that was used to calculate the trim coil sextupole strengths. The program is written in FORTRAN 77 and was tested on the VAX. The input consists of two files. The first file includes the number of trim coils per superperiod; the number of superperiods; position, betatron functions and the phase advance for each trim coil in the first superperiod; the betatron tunes and the circumference. The second file consist of the real and imaginary parts of A_{13}, C_{13}, A_{14} and C_{14}. After the program has run the array sx(*) contains the strength of all the trim coils.

```
      program correctors
c
c       This program calculates the coefficients for each correcting
c       sextupole to determine the resonance strengths the correctors
c       drive.
c
c       We use the thin lens approximation.
c
      implicit real*8 (a-h,o-z)
c
c          anorm   = a normalizing factor for the resonance strengths
c          bx      = the beta function (x direction)
c          by      = the beta function (y direction)
c          cct     = the trim coil package contribution to resonance
c          cfac    = a transformation factor for the resonance strength
c          circ    = the machine circumference
c          cstr    = the real part of the resonance strength to be
c                    excited
c          csup    = the real part of the superperiod contribution
c                    to the resonance
c          maxsize = maximum size for sextupoles per superperiod
c          msxt    = maximum size of sextupole array
c          mux     = the phase advance (x direction, 2*pi included)
c          muy     = the phase advance (y direction, 2*pi included)
c          nct     = the number of trim coil packages per superperiod
c          nres    = the number of resonances
c          nsup    = the number of superperiods
c          nsupmax = maximum superperiod
c          sct     = the trim coil package contribution to resonance
c          sfac    = a transformation factor for the resonance strength
c          snorm   = the coefficient for a given sextupole from one
c                    superperiod to the next
c          sstr    = the imaginary part of the resonance strength to be
c                    excited
c          ssup    = the imaginary part of the superperiod contribution
c                    to the resonance
c          suml    = position of the center of the trim coil package
c          sx      = sextupole array
c          vx      = tune in x direction
c          vy      = tune in y direction
c          wx      = !
c          wy      = !  a resonance => wx*vx + wy*vy = wp
c          wp      = !
c
```

```
      parameter (maxsize=100, nres=4, nsupmax=60, msxt=maxsize*nsupmax,
     1           pi=3.141592653589793d0, twopi=6.2831853071795865d0)
      real*8 mux,muy
      complex*16 ccoef
      character smtrm*4,stype*4
      dimension suml(maxsize),bx(maxsize),mux(maxsize),by(maxsize),
     1          muy(maxsize),wx(nres),wy(nres),wp(nres),csup(nres,
     2          nsupmax),ssup(nres,nsupmax),anorm(nres),cct(nres,
     3          maxsize),sct(nres,maxsize),cstr(nres),sstr(nres),
     4          snorm(nres,nsupmax),sfac(nres),sx(msxt),b(nres),
     5          a(nres,nres),cfac(nres),indx(nres)
c ----- the resonances
      data    wx/ 3.d0, 1.d0, 3.d0, 1.d0/,
     1        wy/ 0.d0, 2.d0, 0.d0, 2.d0/,
     2        wp/13.d0,13.d0,14.d0,14.d0/,
     3     anorm/48.d0,16.d0,48.d0,16.d0/,
     4        sx/msxt*0.d0/
c ----- reading in the lattice parameters
      read(4,10) nct,nsup
   10 format(2i8)
      do 20 i=1,nct
          read(4,30) suml(i),bx(i),zmux,by(i),zmuy
          mux(i)=twopi*zmux
          muy(i)=twopi*zmuy
   30     format(5e16.9)
   20 continue
      read(4,30) circ,vx,vy
c ----- reading the resonance strengths
      read(5,*) (cstr(i),sstr(i),i=1,nres)
c ----- contribution to each resonance by superperiod
      part=twopi/dfloat(nsup)
      write(6,*)
      do 40 i=1,nres
          do 40 j=1,nsup
              arg=wp(i)*part*dfloat(j-1)
              csup(i,j)=dcos(arg)
              ssup(i,j)=dsin(arg)
   40 continue
      do 50 j=1,nsup
          write(6,60) j,(csup(i,j),ssup(i,j),i=1,nres)
   60     format(1x,i4,8f9.6)
   50 continue
c ----- contribution to each resonance from within each superperiod
      conv=twopi/circ
      do 70 i=1,nres
          do 70 j=1,nct
              ccoef=dcmplx(0.d0,bx(j)**(.5d0*wx(i))*by(j)**(.5d0*
     1              wy(i))/(anorm(i)*pi))*cdexp(dcmplx(0.d0,wx(i)*
     2              (mux(j)-conv*vx*suml(j))+wy(i)*(muy(j)-conv*
     3              vy*suml(j))+wp(i)*conv*suml(j)))
              cct(i,j)=dreal(ccoef)
              sct(i,j)=dimag(ccoef)
   70 continue
      write(6,*)
      do 80 j=1,nct
          write(6,60) j,(cct(i,j),sct(i,j),i=1,nres)
   80 continue
c
```

```
c       the sextupoles are modeled by each being equal to its
c       neighbor then for each resonance we solve a set of 4 x 4
c       linear equations to obtain the sextupole sterngths.
c
c ----- the resonance strengths are converted to deal with each superiod
c ----- separately
      do 90 i=1,nres
          cfac(i)=0.d0
          sfac(i)=0.d0
          do 90 j=1,nsup
              snorm(i,j)=csup(i,j)
              cfac(i)=cfac(i)+snorm(i,j)*csup(i,j)
              sfac(i)=sfac(i)+snorm(i,j)*ssup(i,j)
   90 continue
c ----- the linear equation to solve
c ----- the p=13 resonances
      den1=cfac(1)**2+sfac(1)**2
      den2=cfac(2)**2+sfac(2)**2
      b(1)=(cfac(1)*cstr(1)+sfac(1)*sstr(1))/den1
      b(2)=(cfac(1)*sstr(1)-sfac(1)*cstr(1))/den1
      b(3)=(cfac(2)*cstr(2)+sfac(2)*sstr(2))/den2
      b(4)=(cfac(2)*sstr(2)-sfac(2)*cstr(2))/den2
      do 95 i=1,2
          do 95 j=1,4
              a(2*i-1,j)=cct(i,j)+cct(i,j+4)
              a(2*i,j)=sct(i,j)+sct(i,j+4)
   95 continue
      call ludcmp(a,4,nres,indx,dsgn)
      call lubksb(a,4,nres,indx,b)
      do 100 i=1,nsup
          do 100 j=1,4
              karg=8*(i-1)+j
              sx(karg)=snorm(1,i)*b(j)
              sx(karg+4)=sx(karg)
  100 continue
c ----- for p=14 resonances
      den3=cfac(3)**2+sfac(3)**2
      den4=cfac(4)**2+sfac(4)**2
      b(1)=(cfac(3)*cstr(3)+sfac(3)*sstr(3))/den3
      b(2)=(cfac(3)*sstr(3)-sfac(3)*cstr(3))/den3
      b(3)=(cfac(4)*cstr(4)+sfac(4)*sstr(4))/den4
      b(4)=(cfac(4)*sstr(4)-sfac(4)*cstr(4))/den4
      do 105 i=1,2
          do 105 j=1,4
              a(2*i-1,j)=cct(i+2,j)+cct(i+2,j+4)
              a(2*i,j)=sct(i+2,j)+sct(i+2,j+4)
  105 continue
      call ludcmp(a,4,nres,indx,dsgn)
      call lubksb(a,4,nres,indx,b)
      do 110 i=1,nsup
          do 110 j=1,4
              karg=8*(i-1)+j
              sx(karg)=sx(karg)+snorm(3,i)*b(j)
              sx(karg+4)=sx(karg+4)+snorm(3,i)*b(j)
  110 continue
c ----- printing out the value of the sextupoles
      do 120 i=1,nct*nsup
          if (mod(i,2) .eq. 1) then
```

```
            smtrm='+ SF'
            stype='CHRF'
        else
            smtrm='+ SD'
            stype='CHRD'
        endif
        write(7,130) i,stype,sx(i),smtrm
  130   format(4x,'SCT',i2.2,'  : MULTIPOLE, TYPE=',a4,', K2L=',
     1          f19.15,a4)
  120 continue
      end
c**********************************************************************
c      LU Decomposition of a linear system of equations
c**********************************************************************
c
c      W. H. Press, et al; 'Numerical Recipes',(Cambridge University
c          Press, 1986)
c
      SUBROUTINE LUDCMP(A,N,NP,INDX,D)
      implicit real*8 (a-h,o-z)
      PARAMETER (NMAX=100,TINY=1.0d-20)
      DIMENSION A(NP,NP),INDX(N),VV(NMAX)
      D=1.d0
      DO 12 I=1,N
        AAMAX=0.d0
        DO 11 J=1,N
          IF (dABS(A(I,J)).GT.AAMAX) AAMAX=dABS(A(I,J))
11      CONTINUE
        IF (AAMAX.EQ.0.d0) PAUSE 'Singular matrix.'
        VV(I)=1.d0/AAMAX
12    CONTINUE
      DO 19 J=1,N
        IF (J.GT.1) THEN
          DO 14 I=1,J-1
            SUM=A(I,J)
            IF (I.GT.1)THEN
              DO 13 K=1,I-1
                SUM=SUM-A(I,K)*A(K,J)
13            CONTINUE
              A(I,J)=SUM
            ENDIF
14        CONTINUE
        ENDIF
        AAMAX=0.d0
        DO 16 I=J,N
          SUM=A(I,J)
          IF (J.GT.1)THEN
            DO 15 K=1,J-1
              SUM=SUM-A(I,K)*A(K,J)
15          CONTINUE
            A(I,J)=SUM
          ENDIF
          DUM=VV(I)*dABS(SUM)
          IF (DUM.GE.AAMAX) THEN
            IMAX=I
            AAMAX=DUM
          ENDIF
16      CONTINUE
```

```
          IF (J.NE.IMAX)THEN
            DO 17 K=1,N
              DUM=A(IMAX,K)
              A(IMAX,K)=A(J,K)
              A(J,K)=DUM
17          CONTINUE
            D=-D
            VV(IMAX)=VV(J)
          ENDIF
          INDX(J)=IMAX
          IF(J.NE.N)THEN
            IF(A(J,J).EQ.0.)A(J,J)=TINY
            DUM=1.d0/A(J,J)
            DO 18 I=J+1,N
              A(I,J)=A(I,J)*DUM
18          CONTINUE
          ENDIF
19      CONTINUE
        IF(A(N,N).EQ.0.d0)A(N,N)=TINY
        RETURN
        END
C-----------------------------------------------------------------------
        SUBROUTINE LUBKSB(A,N,NP,INDX,B)
        implicit real*8 (a-h,o-z)
        DIMENSION A(NP,NP),INDX(N),B(N)
        II=0
        DO 12 I=1,N
          LL=INDX(I)
          SUM=B(LL)
          B(LL)=B(I)
          IF (II.NE.0)THEN
            DO 11 J=II,I-1
              SUM=SUM-A(I,J)*B(J)
11          CONTINUE
          ELSE IF (SUM.NE.0.) THEN
            II=I
          ENDIF
          B(I)=SUM
12      CONTINUE
        DO 14 I=N,1,-1
          SUM=B(I)
          IF(I.LT.N)THEN
            DO 13 J=I+1,N
              SUM=SUM-A(I,J)*B(J)
13          CONTINUE
          ENDIF
          B(I)=SUM/A(I,I)
14      CONTINUE
        RETURN
        END
```

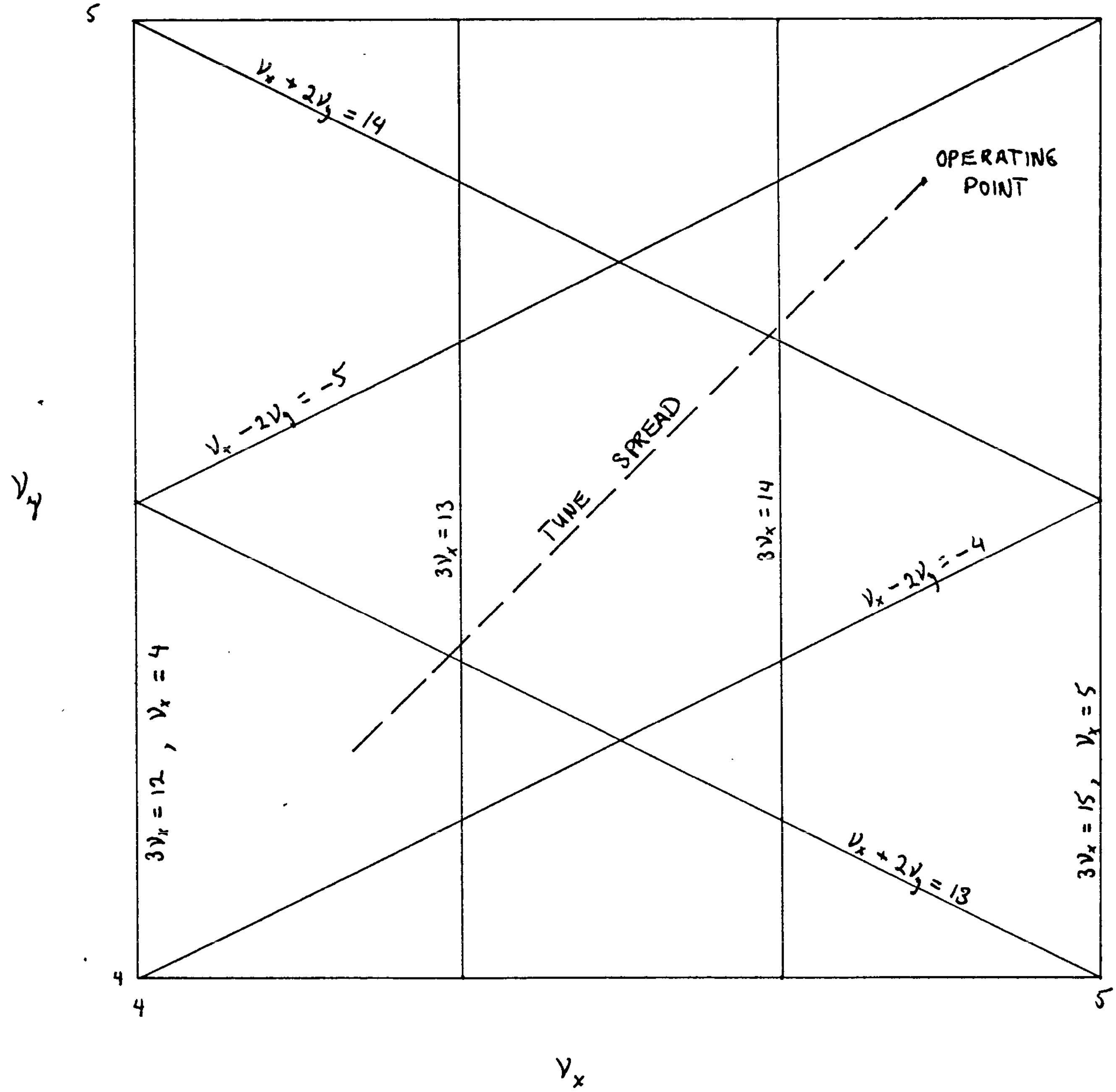

Fig. 1 Tune Diagram

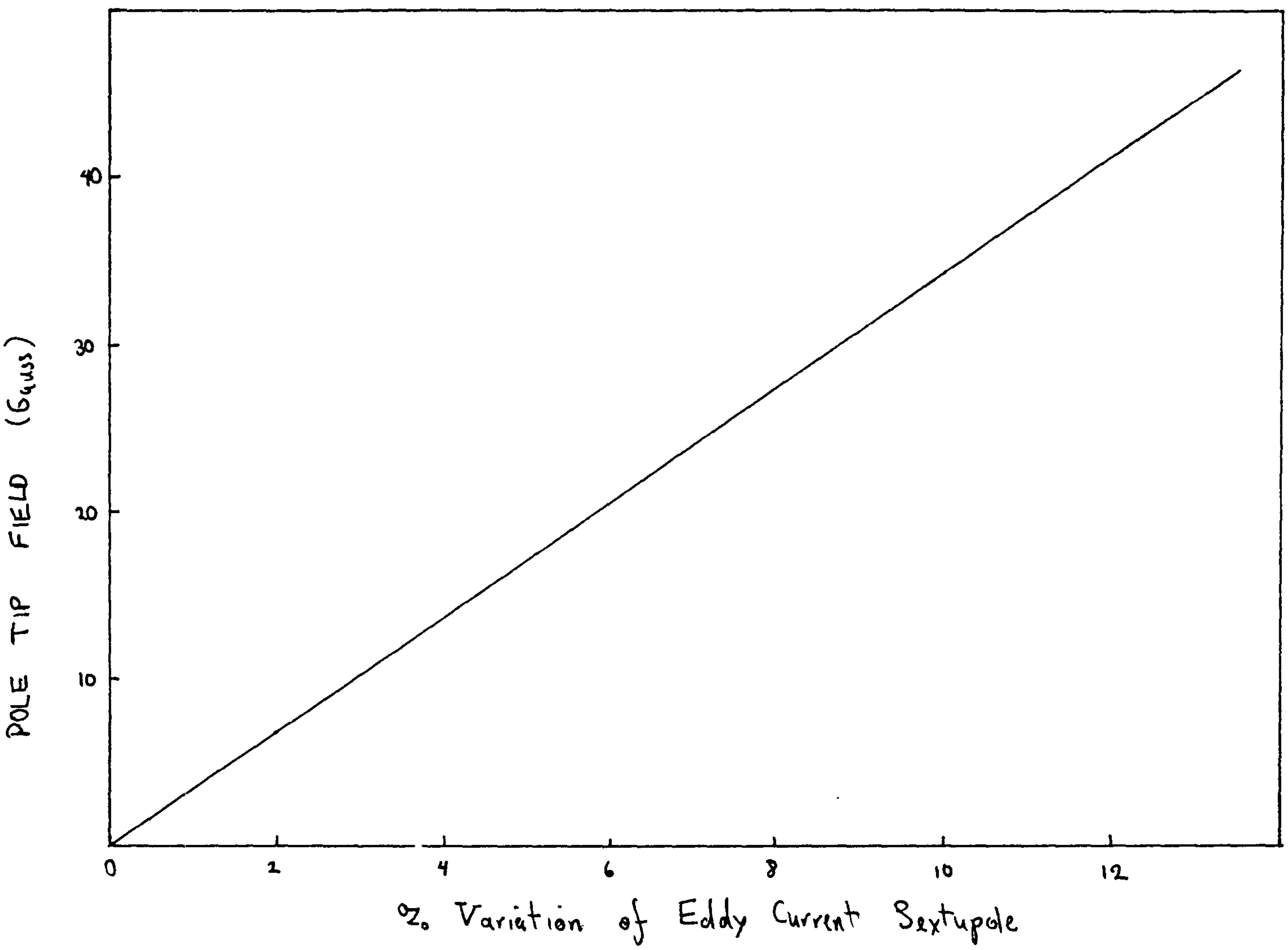

Fig. 2 Pole tip strength versus random sextupole strength