

# Luminosity Calculation From Known Beam Functions

Galekhutle Marang

*Electrical Engineering, Wayne State University, Detroit, MI, 48202*

(Dated: August 16, 2003)

Luminosity is a measure of the rate of collisions over a given area in a beam/beam interaction. Once calculated, this value can be used to make certain inferences about the machine being used. Since the higher the luminosity is, the more particles are involved in any particular collision; the luminosity can also be a measure of how efficiently the storage ring settings are in inducing the beams to collide. Additionally if we calculate the luminosity using data regarding beam characteristics, then further information about how storage ring parameters and individual storage ring elements affect the beam can be derived. It is the goal of this project to create a program using Fortran90 that performs such a calculation.

## I. INTRODUCTION

Presently, inside the CESR storage ring, measurements of luminosity are not taken directly from the beams, but are calculated using information collected from the detectors surrounding the interaction region. These detectors can identify, count, and track particles produced by the collision of the positron and electron beams. Using this data, the number of particles that would need to interact between the two beams in order to produce the observed effect can be determined. This method of calculation may be accurate to less than a percent and has proven to be reliable, but using this method it is difficult to trace changes in the luminosity values back to their causes in the storage ring.

It is understood how storage ring settings affect a beams characteristics: size, shape, speed, distribution, position and crossing angle. These characteristics, along with other factors, are responsible for the luminosity values produced. However, because information is not collected about these characteristics in the current calculation method, the link between luminosity and beam characteristics has not been fully explored. By analyzing these characteristics as the beam approaches and passes through the IR, this link may be defined. Ultimately, storage ring parameters may be directly associated with luminosity values they produce.

In this experiment we hope to create a program **Test Lum Calc**, using Fortran90, that will calculate luminosity directly from beam characteristics / behavior that are known to occur in the IR. This program will calculate luminosity using a storage ring simulation for a given lattice ( storage ring settings.) The effects of coupling and the hour glass effect will be taken into account by the program. When functioning, this program can be used to fill the information gap between beam characteristics and resultant luminosity.

## II. BEAM CHARACTERISTICS AND LUMINOSITY

The characteristics of the electron and positron beams are determined by storage ring elements. The beams enter the storage ring travelling in opposite directions. Bending magnets force the beams to travel in a circular motion around the ring while other magnets

move the beams away from each other at all points except inside the detector, ensuring this is the only collision point. The angle at which the beams enter the detector and eventually collide is controlled in this way. According to the simplified luminosity formula of equation 1

$$L = \frac{N_1 N_2 f}{4A} \quad (1)$$

the smaller the area ( $A$ ) over which the beams collide, the larger the luminosity can be. In a head on collision the beams interact over the entire length of a beam, but in a collision with a crossing angle the beams pass through one another over a smaller region. A small horizontal crossing angle is used presently in CESR.

To use equation 1 we not only need to know the area over which the beams collide, but we also need to know the number of particles in each beam ( $N_1$  and  $N_2$ ) and the collision frequency of the beams. For our calculations we assume that the number of particles in each beam is  $10^{10}$ , grouped in 45 bunches. Radio frequency cavities are used to accelerate the beams as they move through the ring and help to determine the rotational frequency of the beam. The collision frequency ( $f$ ) can be determined by multiplying the rotational frequency of the beam by the number of bunches in the beam.

There are other beam characteristics affecting luminosity that are not directly referred to by equation 1. The size of the beam is one such characteristic and is controlled by the focusing elements (quadrupole magnets) working in the horizontal and vertical dimensions and by the RF cavity that, in addition to accelerating the beam, compacts the beam longitudinally. Despite being stretched and compressed in three dimensions the beam maintains a gaussian distribution in each dimension. This uniformity allows us to analyze the motion and behavior of the central particle in the beam (taken to be the origin of the horizontal ( $X$ ), vertical ( $Y$ ), and longitudinal ( $Z$ ) axes) and determine the behavior of the other beam particles in relation to it. The square root of the Beta function defines two other beam characteristics, the size and position of the beam envelope as it travels through the ring.

### III. BEAM EFFECTS AND LUMINOSITY

Coupling between the  $X$  and  $Y$  dimensions and the hour glass beam effects can have a significant impact on the accuracy of calculated luminosity values. Coupling (where changes in the  $X$  dimension of the beam are not independent of the  $Y$  dimension) can be accounted for mathematically in the distribution function by using eigen coordinates (refer to Section IV.) The hour glass effect causes the beam to expand in the  $Y$  dimension just before and just after the interaction region. To take this effect into account we must analyze the  $Y$  dimension over a large range, much larger than the range of the  $X$  and  $Y$  dimensions, so that the distribution function can accurately describe the beam.

### IV. CALCULATION AND PROGRAM DEVELOPMENT

In order to calculate luminosity we must first determine the probability of particles in each beam colliding. Using this probability we can determine the number of particles from both beams that will collide with one another. The major challenge in creating the program

**Test Lum Calc** was in building the distribution function, equation 2.

$$f(x, x'y, y', z, \delta) = K \exp\left[\frac{-1}{2}\vec{p}^T \Sigma \vec{p}\right] \quad (2)$$

The distribution function describes the distribution of particles within the beam in six phase space coordinates: x - horizontal axes, x' - angle from horizontal, y - vertical, y' angle from vertical, z - longitudinal,  $\delta$ - fractional energy deviation. to calculate the function we use  $\vec{P}$ , the phase space coordinate vector of a particle;  $\Sigma$ , a six dimensional emittance matrix made up of emittance sub-matrixes x,y,and L; and the normalizing factor K. When coupling is taken into account, the formula becomes

$$f(x, x'y, y', z, \delta) = K \exp\left[\frac{-1}{2}\vec{p}^T (V^T)^{-1} \Sigma V^{-1} \vec{p}\right] \quad (3)$$

where V is the transport matrix and  $\Sigma$  is now made up of eigen sub-matrixes u, v, L.

In order to make the calculation of the distribution of the entire beam easier to manage, we divide the beam into smaller six space volumes creating a mesh. The distribution calculated in each of these mesh points is then added together for the total distribution. The smaller the volumes we analyze, the finer the mesh will be, and the more calculations will take place providing a more accurate value.

In **generate mesh**, a subroutine of the program **Test Lum Calc** the values used to fill in matrix V are provided directly from the simulation. The values for the  $\Sigma$  matrix must be calculated and placed in position using the values  $\gamma, \alpha, \text{ and } \beta$  provided by the simulation. The value of  $V^{T^{-1}} * \Sigma_{u,v,l} * V^{-1}$  is stored for storage ring positions (s) at different energies. The values of s stored represent the storage ring positions just before the IR, in the IR and just after the IR. S has its zeroth position at the interaction point.

The calculation for the distribution function is continued in the sub-routine **f unnormal** called by the sub-routine **Lum Calc**. Inside **Lum Calc** we nested seven loops representing of all the possible mesh positions at each point in S. (From outer most loop to inner most loop: s,z,x,y,  $\delta$ , x', y') Since the phase space coordinate vector  $\vec{p}$  must be calculated at each position, we call **f unnormal** inside the innermost loop. Inside **f unnormal** we calculate  $\vec{p}$  and complete the exponential of the distribution function.  $\vec{P} = \text{the mesh position} - \text{any displacement we may choose to include} - \text{the orbit position at the same position in S and at the same energy} + \text{any displacement contribution from the angles}$ . All the necessary values for the exponential have been provided at this point so we perform  $\exp\left[\frac{-1}{2}\vec{p}^T (V^T)^{-1} \Sigma V^{-1} \vec{p}\right]$  and send the value back to **Lum Calc**. That value is summed with the value obtained for each mesh position and the normalizing value K is computed.  $K = 1/\text{sum for each position in s}$

With the distribution function complete, it can then be plugged into the larger luminosity formula **L**.

$$L = N_1 N_2 f_{bb} \int d^6 V_1 \int ds_1 \int d^6 V_2 \int ds_2 f_1 f_2 * \sigma(s_1 + s_2) \sigma(s_1 + z_1 - s_2 + z_2) \sigma(x_1 - x_2) \sigma(y_1 - y_2) \quad (4)$$

In this equation  $N_1$  and  $N_2$  are the number of particles in the two beams,  $f_{bb}$  is the collision frequency of the two beams, and  $f_1$  and  $f_2$  are the distribution functions of the two beams. The  $\sigma$  terms at the end of the equation ensure that the same position in beam 1 and beam 2 are being analyzed as the beams move through one another; they define the area being analyzed. It is important to note that gives the probability of collision for particles in beam 1, as  $f_2 * \int d^6 V_2$  does for beam 2. Multiplied together we get the probability of the particles

of each beam colliding with one another in a given area.  $\int ds_1$  and  $\int ds_2$  tell us to sum the probability of collision for both beams over the trajectory of both beams.

We must carry out the luminosity function inside **Lum Calc**. We start by multiplying the sum of the distribution function for beam 1 by the sum of the distribution function for beam 2 then add this product to itself for each y loop. This gives us the probability of particles colliding in beam 1 multiplied by the probability of particles colliding in beam 2. By performing the calculations of the distribution function and the probability of collision at the appropriate places within the loop structure we ensure that we are comparing the beams at the same position as the beams move through each other (fulfilling the role of the  $\sigma$  terms in the Luminosity formula.) By calculating the probability of collision for both beams inside the S loop we ensure that the values are summed over the trajectory of both beams. We now leave the outermost loop and multiply our probability value by the number of particles in beams 1 and 2 (assumed to be  $10^{10}$  for both beams) and by the collision frequency (number of bunches in each beam multiplied by the rotational frequency). The resultant value is the luminosity.

## V. TESTING

For preliminary testing we compared the luminosity calculated by **Test Lum Calc** to luminosity calculated using the storage ring for a given set of parameters. The values were close enough to suggest that the program was functioning properly and calculating reasonable values. We did not require total agreement between the two measurements since the efficiency of the program is not yet known. We also built graphing capabilities into the program so the distribution of the beam in the X, Y, and Z dimensions could be observed. It was apparent from the graphs that the distributions were in the Y dimension were not overlapping completely for the two beams at the IR. Because the y dimension of the beam becomes so small just as it enters the IR without changing its distribution, any misalignment of the two beams in the y dimension results in large errors in the luminosity calculated. To correct this misalignment, we made adjustments to the y term of the displacement vector until the desired overlap was achieved and luminosity was maximized. It was found that the displacement required to achieve maximum luminosity was not the same as the mesh became finer. This is not surprising since as the mesh becomes finer, more points are being analyzed and luminosity can be calculated more accurately, table I.

The behavior of the luminosity was observed by moving the electron beam center back and forth in the z dimension across the interaction region and seeing how the luminosity changed. The bunch length is set to .01m, with 45 bunches being used in each beam and a circulation frequency of .39012MHz. The expected result was that there be a parabolic range of values having a maximum when the z displacement equal to zero. As we moved the electron's z displacement from -6mm to 7mm a parabolic range of values was observed, but the maximum was at 5mm instead of 0mm, table II. Given the possibility that the program was accessing areas outside the mesh and calculating wrong values, we used the same displacement but varied both beams such that at -2mm positron displacement and 2mm electron displacement the overall displacement is the same as 0mm positron displacement and 4mm electron displacement as tested before. When both beams were displaced similar values were achieved, maximum at -2.5 positron displacement and 2.5 electron displacement, table III. It is believed that the luminosity calculation is correct, but that the beams may not be colliding at the position  $s = 0$  as believed.

The finer the mesh becomes the more points are analyzed, increasing both the accuracy of the calculation and the time it takes to perform it. Because the time it takes to run is a major handicap of the program, part of the testing involves modifying the mesh parameters to see how the luminosity calculation improves or declines with the aim of finding a minimal mesh size that does not sacrifice more than 1 per cent precision in luminosity. The coarseness of the mesh in each dimension is defined by two numbers, **num sigma** (the number of sigma we will use in either direction from zero to define the beam) and **num per sigma** (the number of points that will be analyzed inside each sigma) these numbers are multiplied and then doubled to determine the number of mesh points to be analyzed in both directions of a particular dimension. Thus far the changes in these two mesh parameters have produced significant differences in the luminosity calculated. There is an increase of 28 per cent in the luminosity values when comparing the values num per sigma= 1 to those of num per sigma= 2; and of 46 per cent when comparing num per sigma = 2 to num per sigma = 3, where num sigma x = 3, num sigma y = 18 , num sigma z = 3 for all calculations. There is also an increase of 40 per cent in the luminosity values when comparing the value at num per sigma= 2 where num sigma x = 3, num sigma y = 18, num sigma z = 3 to that of num per sigma = 2 where num sigma x = 4, num sigma y = 24, num sigma z = 4; table IV. These comparisons suggest that the minimal mesh size needed may be quite large. Considering at num per sigma = 3 the program takes several hours to run, there will need to be changes made to the program to make it run faster.

The last test we were able to perform compared our luminosity value to the simplified calculation of equation 1. The beam parameter were specified as:  $Beta * x = 0.5886$ ,  $Beta * y = 0.0141$ , and  $Sigma z = 5mm$  .) The expected value was  $1.07 \times 10^{31}$  . Using a very course mesh the program calculated a value of  $0.9158940 * 10^{31} / cm^2 / sec$ , 85 per cent of the expected value. This value is very close considering the mesh being used. There is a chance that this value may be over estimated, however, as the emittance values for the program's calculation were not constrained to those of the simplified calculation.

## VI. TABLES

TABLE I: Maximum Luminosity (Lum ( $*10^30/cm^2/sec$ ))

(Positron, Electron) dz (mm)	<i>NumPerSigma</i> = 1	<i>NumPerSigma</i> = 2
(.009, -.009)	1.505711	—
(.010, -.010)	1.583742	5.303195
(.011, -.011)	1.615743	5.476451
(.012, -.012)	1.600056	5.485400
(.013, -.013)	—	5.335505

TABLE II: Electron dz Variation

Electron dz (mm)	Luminosity ( $*10^30/cm^2/sec$ )
7	1.638173
6	1.640212
5	1.640640
4	1.639372
3	1.636323
2	1.631407
1	1.624551
0	1.615743
-1	1.605011
-2	1.592440
-3	1.578130
-4	1.562237
-5	1.544920
-6	1.526322

TABLE III: Electron/Positron dz Variation

Electron dz (mm)	Positron dz (mm)	Luminosity ( $*10^30/cm^2/sec$ )
-3	3	1.644372
-2.5	2.5	1.643938
-2	2	1.641727
-1	1	1.632075
1	-1	1.593064
2	-2	1.564350
3	-3	1.529865

TABLE IV: Optimal Mesh Size

Num Per Sigma	Num Sigma X (mm)	Luminosity ( $*10^30/cm^2/sec$ )
1	2	0.129257
2	2	0.622264
3	2	1.434044
1	3	0.613783
2	3	2.220041
3	3	4.791465
1	4	1.643938
2	4	5.606051
3	4	—

## VII. CONCLUSIONS

We were able to build a program that calculates luminosity based on the behavior and motion of the positron and electron beams. The program produces reasonable luminosity values for given storage ring parameters, beam parameters, and beam displacements when run using a coarse mesh. A key test to perform in the future will involve finding the optimal mesh size for Test Lum Calc to run with. Test values for luminosity should be generated with this mesh; ensuring the program is performing the calculations correctly and to allow for a more direct comparison with luminosity as calculated using the storage ring detectors. Once the program has proven to run correctly, the objective will be to reduce the time it takes the program to run enabling the program to run using data from live storage ring events.

## VIII. ACKNOWLEDGMENTS

I would like to thank Michael G. Billing, of Cornell University, for his patience and guidance in pursuing this Research Experience for Undergraduates project and for increasing my understanding of particle physics exponentially. Also thanks to Rich Galik, Cornell University, for organizing the Cornell University REU program. (It was very enjoyable.) Special thanks to Giovanni Bonvicini, of Wayne State University, for organizing the Wayne State Universtiy REU program; for providing valuable background information that made the transition from classroom to lab so much easier; and for his continued counsel throughout the program.

This work was supported by the National Science Foundation REU grant PHY-0101649 and research co-operative agreement PHY-9809799.

- 
- [1] Billing, Michael. *Geometrical Calculation of Luminosity Part I: General Formulation*. (2002)  
 [2] Sands, Matthew. *The Physics of Electron Storage Rings: An Introduction* (1970)